

Área de Conocimiento de Tecnología de la
Información y Comunicación

Desarrollo de un sistema web para la gestión educativa del Proyecto “Sara María Parrales” (PROYSMAPA).

**Trabajo monográfico para optar al título
de Ingeniero en Computación**

Elaborado por:

Br. Nestor Gabriel
Aguirre Canales
Carnet: 2020-0479U

Br. Samuel Ulises
Martínez Martínez
Carnet: 2011-37305

Tutor:

Msc. Luis Eduardo
Chávez Mairena

RESUMEN

El presente trabajo monográfico, tiene como objetivo el desarrollo de un sistema web para la gestión educativa del Proyecto Comunitario Sara María Parrales (**PROYSMAPA**), ubicado en la comunidad de San Jacinto, municipio de Telica.

Esta iniciativa está orientada a brindar atención integral a niños y adolescentes en situación de riesgo, actualmente este proyecto enfrenta diversas limitaciones debido al uso de métodos manuales y herramientas no integradas en su administración.

Con el fin de superar esta problemática, se plantea una solución tecnológica que permita centralizar y automatizar los procesos administrativos claves, mejorando así la gestión de beneficiarios, la planificación de actividades y el uso eficiente de los recursos disponibles. El sistema será desarrollado bajo la metodología ágil Scrum, utilizando tecnologías como ASP.NET Core 8 y SQL Server, lo que asegurará un producto escalable, eficiente y fácil de usar para el personal del centro.

Entre los principales beneficios esperados se encuentran: una mayor eficiencia operativa, la reducción de errores, el fortalecimiento de la toma de decisiones y la creación de un entorno más organizado para el seguimiento de los beneficiarios. Como desafío inicial, se reconoce el proceso de adaptación del personal al nuevo sistema, así como la necesidad de capacitación básica en el uso de herramientas digitales.

En conjunto, esta propuesta busca fortalecer la sostenibilidad del proyecto y ampliar su impacto positivo en la comunidad, facilitando el cumplimiento de su misión educativa, moral y espiritual a través del uso efectivo de la tecnología.

ÍNDICE DE CONTENIDO

I. INTRODUCCIÓN	1
II. ANTECEDENTES.....	2
III. JUSTIFICACIÓN	3
IV. OBJETIVOS.....	4
4.1. OBJETIVO GENERAL.....	4
4.2. OBJETIVOS ESPECIFICOS	4
V. MARCO TEÓRICO	5
5.1. EDUCACIÓN INTEGRAL	5
5.2. SISTEMAS DE GESTIÓN DE BENEFICIARIOS	5
5.3. TECNOLOGÍAS WEB.....	5
5.4. INTERNET	6
5.5. UML	6
5.6. HTML	6
5.7. CSS3.....	6
5.8. JAVASCRIPT.....	7
5.9. NET CORE 8.....	7
5.10. RAZOR	7
5.11. AJAX	7
5.12. JSON	8
5.13. METODOLOGÍAS ÁGILES	8
5.14. CONCEPTOS CLAVE DE SCRUM	8
5.15. FUNCIONAMIENTO DE SCRUM	10
5.16. BENEFICIOS DE SCRUM	10
VI. DISEÑO METODOLÓGICO	12
6.1. ROLES SCRUM.....	12
6.2. ARTEFACTOS SCRUM.....	12
6.3. EVENTOS SCRUM	13

6.4.	SPRINTS PARA PROYSMAPA.....	14
6.5.	PROCESO SCRUM PARA PROYSMAPA	20
VII.	ANÁLISIS Y PRESENTACIÓN DE RESULTADOS	22
7.1.	ESTUDIO DE FACTIBILIDAD.....	22
7.2.	FACTIBILIDAD OPERATIVA	22
7.3.	FACTIBILIDAD TÉCNICA	23
7.4.	FACTIBILIDAD ECONÓMICA	24
7.5.	REQUERIMIENTOS DEL SISTEMA.....	27
7.6.	CASOS DE USO DEL SISTEMA.....	32
7.7.	DISEÑO DEL MODELO ENTIDAD RELACIÓN	53
7.8.	DISEÑO DE DIAGRAMAS DE ACTIVIDADES.....	55
7.9.	DISEÑO DE LA INTERFAZ DE USUARIO DEL SISTEMA.....	67
7.10.	CODIFICACIÓN DEL SISTEMA.....	70
7.11.	PRUEBAS UNITARIAS	73
7.12.	IMPLANTACIÓN DEL SISTEMA.....	78
VIII.	CONCLUSIONES.....	80
IX.	RECOMENDACIONES.....	82
X.	BIBLIOGRAFIA.....	84
XI.	ANEXOS	86
	SOLUCIÓN A LOS SPRINT	87
	ENTREVISTA	114
	CARTAS DE SOPORTE DEL PROYECTO.....	116

ÍNDICE DE TABLAS

Tabla 1: Herramientas para el sistema.....	24
Tabla 2: Costo de los desarrolladores	25
Tabla 3: Requisitos para Servidor Web	26
Tabla 4: Costo Total del Proyecto	26
Tabla 5: Requerimientos Profesiones.....	28
Tabla 6: Requerimientos Aulas.....	28
Tabla 7: Requerimiento Ciclo Académico	28
Tabla 8: Requerimientos Cursos	29
Tabla 9: Requerimientos Turnos	29
Tabla 10: Requerimientos Matricula	29
Tabla 11: Requerimientos Personas.....	30
Tabla 12: Requerimientos Tipo de Identificación	30
Tabla 13: Requerimientos Asistencia Escolar	30
Tabla 14: Requerimientos Notas Escolares.....	31
Tabla 15: Requerimientos Consulta Médicas y Psicológicas.....	31
Tabla 16: Registrar Profesión	32
Tabla 17: Editar Profesión	32
Tabla 18: Dar baja profesión	33
Tabla 19: Asociar profesión a personas.....	33
Tabla 20: Validar unicidad de la profesión	33
Tabla 21: Exportar lista de profesiones	33
Tabla 22: Registrar aula.....	34
Tabla 23: Validar nombre único de aula.....	34
Tabla 24: Visualizar aulas activas	34
Tabla 25: Editar datos de un aula.....	35
Tabla 26: Dar de baja un aula.....	35
Tabla 27: Exportar información de aulas.....	35
Tabla 28: Registrar ciclo académico	36
Tabla 29: Validar nombre único del ciclo académico	36
Tabla 30: Visualizar lista de ciclos académicos	36

Tabla 31: Editar ciclo académico.....	37
Tabla 32: Dar de baja ciclo académico.....	37
Tabla 33: Nombre Exportar información de ciclos académicos.....	37
Tabla 34: Registrar curso académico	38
Tabla 35: Validar nombre único de curso	38
Tabla 36: Visualizar lista de cursos activos	38
Tabla 37: Editar información de curso	39
Tabla 38: Dar de baja curso	39
Tabla 39: Exportar información de cursos	39
Tabla 40: Registrar turno	40
Tabla 41: Validar nombre de turno único	40
Tabla 42: Visualizar lista de turnos activos.....	40
Tabla 43: Editar turno.....	41
Tabla 44: Dar de baja turno.....	41
Tabla 45: Exportar información de turnos.....	41
Tabla 46: Registrar matrícula.....	42
Tabla 47: Validar unicidad de matrícula	42
Tabla 48: Listar matrículas	42
Tabla 49: Editar matrícula.....	43
Tabla 50: Dar de baja matrícula	43
Tabla 51: Exportar información de matrículas.....	43
Tabla 52: Registrar persona.....	44
Tabla 53: Tipificar persona	44
Tabla 54: Validar unicidad del documento de identidad	44
Tabla 55: Visualizar lista de personas	44
Tabla 56: Editar datos de una persona	45
Tabla 57: Dar de baja a una persona	45
Tabla 58: Exportar información de personas.....	45
Tabla 59: Registrar tipo de identificación	46
Tabla 60: Validar unicidad del tipo de identificación.....	46
Tabla 61: Visualizar listado de tipos de identificación	46

Tabla 62: Editar tipo de identificación	46
Tabla 63: Dar de baja tipo de identificación	47
Tabla 64: Exportar información de tipos de identificación	47
Tabla 65: Registrar asistencia de estudiantes.....	47
Tabla 66: Validar asistencia por día.....	48
Tabla 67: Visualizar lista de estudiantes para asistencia	48
Tabla 68: Editar asistencia	48
Tabla 69: Exportar información de asistencia	48
Tabla 70: Registrar notas escolares	49
Tabla 71: Validar duplicidad de notas.....	49
Tabla 72: Visualizar notas escolares	49
Tabla 73: Editar notas escolares	50
Tabla 74: Exportar notas escolares.....	50
Tabla 75: Registrar consulta médica o psicológica	50
Tabla 76: Validar unicidad de nombre de consulta.....	51
Tabla 77: Visualizar listado de consultas	51
Tabla 78: Editar consulta	51
Tabla 79: Exportar información de consultas	51
Tabla 80: Casos de Prueba – Profesiones	73
Tabla 81: Casos de Prueba – Aulas	74
Tabla 82: Casos de Prueba – Ciclo Académico.....	74
Tabla 83: Casos de Prueba – Cursos.....	74
Tabla 84: Casos de Prueba - Turno.....	74
Tabla 85: Casos de Prueba - Matriculas	75
Tabla 86: Casos de Prueba - Personas	75
Tabla 87: Casos de Prueba - Tipo de Identificación.....	75
Tabla 88: Casos de Prueba – Asistencia	75
Tabla 89: Casos de Prueba - Notas.....	76
Tabla 90: Casos de Prueba - Consultas Médicas y Psicológicas	76
Tabla 91: Código de Prueba Unitaria.....	77

ÍNDICE DE FIGURAS

Figura 1: Metodología SCRUM.....	13
Figura 2: Modelo Entidad Relación	54
Figura 3: Diagrama de Actividades - Login.....	55
Figura 4: Diagrama de Actividades – Profesiones.....	56
Figura 5: Diagrama de Actividades – Aulas.....	57
Figura 6: Diagrama de Actividades – Ciclo Académico	58
Figura 7: Diagrama de Actividades – Cursos	59
Figura 8: Diagrama de Actividades – Turnos	60
Figura 9: Diagrama de Actividades – Matriculas.....	61
Figura 10: Diagrama de Actividades – Registro Persona.....	62
Figura 11: Diagrama de Actividades – Tipo Identificación.....	63
Figura 12: Diagrama de Actividades – Asistencia Escolar.....	64
Figura 13: Diagrama de Actividades – Notas	65
Figura 14: Diagrama de Actividades – Consultas Médicas o Psicológicas	66
Figura 15: Login del Sistema.....	67
Figura 16: Menú Principal del Sistema.....	68
Figura 17: Formularios del Sistema	68
Figura 18: Tablas del Sistema	69
Figura 19: Modelo MVC	71

ÍNDICE DE ANEXO

Anexo 1: Análisis de Requerimientos y Diseño Inicial	87
Anexo 2: Configuración del Entorno y Arquitectura del Sistema.....	90
Anexo 3: Desarrollo Inicial de Módulos e Interfaces	93
Anexo 4: Implementación de Funcionalidades Clave	96
Anexo 5: Funcionalidades Adicionales y Revisión del Diseño	99
Anexo 6: Módulos de Usuario y Seguridad.....	101
Anexo 7: Finalización de Funcionalidades de Usuario y Pruebas de Usabilidad.	103
Anexo 8: Funcionalidades Avanzadas y Pruebas del Sistema.....	105
Anexo 9: Intercambio de Datos con JSON	106
Anexo 10: Ajustes Finales y Pruebas de Aceptación.....	108
Anexo 11: Preparación para el Despliegue	110
Anexo 12: Despliegue en Producción y Monitoreo.....	112
Anexo 13: Entrevista a PROYSMAPA.....	114
Anexo 14: Carta de Entrega y Soporte de Proyecto	116

I. INTRODUCCIÓN

Los centros de atención integral para niños y adolescentes en riesgo se han convertido en un recurso crucial en muchas comunidades, brindando apoyo no solo en el ámbito moral y espiritual, sino también en su desarrollo personal y técnico. En este contexto, iniciativas como el Proyecto “Sara María Parrales”, fundado en 2020 en la comunidad de San Jacinto, Telica, han emergido como pilares fundamentales para el futuro de estos jóvenes.

Dado el crecimiento del proyecto, junto con los desafíos administrativos que ha enfrentado, la implementación de un sistema de gestión educativo se ha vuelto una necesidad imperativa. Este sistema facilitó la gestión administrativa y educativa del centro.

El presente documento tiene como propósito brindar la información necesaria para el trabajo monográfico “Desarrollo de un sistema web para la gestión educativa del Proyecto “Sara María Parrales” (PROYSMAPA)”, con el fin de ofrecer una visión detallada de los aspectos técnicos y operativos que se abordaron. Este sistema permite continuar su labor con mayor eficiencia y alcance, asegurando un futuro sostenible para los beneficiarios que forman parte de esta comunidad.

II. ANTECEDENTES

En Nicaragua, el Proyecto “Sara María Parrales”, a pesar de su crecimiento y los logros alcanzados, enfrenta importantes desafíos. Actualmente, la falta de un sistema de gestión administrativa y educativa de los beneficiarios ha ralentizado su progreso. La gestión manual de la información no solo afecta la eficiencia operativa, sino que también genera un ambiente de incertidumbre y dificulta la toma de decisiones estratégicas. En el año 2020 se realizaron esfuerzos por optimizar la administración a través de sistemas manuales; sin embargo, estos no se completaron debido a la ausencia de mecanismos eficientes que logran agilizar dichos procesos. La propuesta actual para desarrollar el sistema web PROYSMAPA surge como una respuesta a esta necesidad, buscando optimizar la eficiencia y calidad en la gestión, garantizando que el proyecto siga cumpliendo con su misión de ofrecer un entorno seguro y de alta calidad para todos sus beneficiarios.

De manera complementaria, en el ámbito internacional existen antecedentes que refuerzan la importancia de la digitalización en la gestión educativa. Un ejemplo relevante lo constituye el Sistema de Información para la Gestión Escolar (SIGED) en México, impulsado por la Secretaría de Educación Pública (SEP). Antes de su implementación, la administración de datos escolares se realizaba de forma manual y dispersa en cada centro educativo, lo que generaba retrasos, duplicidad de información y limitaciones para la planificación académica. Con la puesta en marcha del SIGED, se logró centralizar la información de alumnos, docentes y planteles en una plataforma digital, permitiendo un mejor control administrativo, mayor transparencia en la gestión y la posibilidad de diseñar políticas educativas basadas en datos confiables. Este antecedente internacional refleja cómo la migración de procesos manuales hacia un sistema web integral puede transformar la eficiencia operativa y garantizar un impacto positivo en la calidad educativa, experiencia que guarda una relación directa con los objetivos que persigue el Proyecto “Sara María Parrales” en Nicaragua.

III. JUSTIFICACIÓN

El Proyecto “Sara María Parrales”, desde su fundación ha enfrentado múltiples desafíos en la gestión de su centro, el crecimiento del proyecto ha evidenciado la necesidad urgente de mejorar la eficiencia de sus procesos administrativos.

Actualmente, la gestión del proyecto se realiza de manera manual o mediante herramientas no integradas como hojas de cálculo o documentos físicos, lo que genera varios problemas, como pérdida de información, errores administrativos y dificultades en la coordinación de actividades. Estos problemas afectan no solo la calidad de los servicios que ofrecen, sino también su capacidad para cumplir sus objetivos de manera eficiente.

Ante estos retos, se llevó a cabo el “Desarrollo de un sistema web para la gestión educativa del Proyecto Sara María Parrales (PROYSMAPA)”, que ofrece soluciones a los principales problemas detectados, tales como:

- **Eficiencia administrativa:** El sistema automatizará los procesos, permitiendo al personal centrarse en tareas de mayor impacto para los beneficiarios.
- **Gestión de beneficiarios:** Centralizará la información de los beneficiarios atendidos, facilitando el seguimiento adecuado de su progreso.
- **Coordinación de actividades:** Permitirá una planificación más efectiva de talleres, actividades y formaciones técnicas, mejorando la participación de los beneficiarios y el uso de los recursos.

Con la implementación de PROYSMAPA, se espera generar un cambio significativo en la gestión del proyecto, permitiendo una mejor administración del centro, que a su vez contribuirá directamente al desarrollo sostenible de los beneficiarios en riesgo que forman parte de su misión.

IV. OBJETIVOS

4.1. OBJETIVO GENERAL

Desarrollar un sistema web para la gestión educativa del proyecto comunitario Sara María Parrales PROYSMAPA.

4.2. OBJETIVOS ESPECIFICOS

- Analizar los requerimientos funcionales y no funcionales del sistema web, mediante la recopilación de información con los involucrados claves.
- Diseñar un modelo que cumpla con los requerimientos definidos, utilizando la metodología ágil SCRUM y el Lenguaje Unificado de Modelado (UML).
- Codificar el sistema web utilizando el framework ASP.NET Core 8 y SQLServer como gestor de bases de datos.
- Realizar pruebas unitarias utilizando el framework MSTest, incorporado en .NET Core 8, para identificar, corregir y prevenir posibles errores de cada componente del sistema.
- Implementar el sistema web en un entorno de producción local, asegurando su accesibilidad y operatividad para todos los usuarios dentro de la comunidad educativa.

V. MARCO TEÓRICO

El presente marco teórico tiene como objetivo ofrecer los conceptos claves que sustentan el desarrollo e implementación de un sistema web para el Proyecto Educativo Sara María Parrales. A continuación, se detallan los principales conceptos involucrados en la solución propuesta:

5.1. EDUCACION INTEGRAL

La educación integral es un enfoque que abarca más allá del desarrollo académico, promoviendo el crecimiento personal de los estudiantes en las áreas emocional, social, física y espiritual. Este concepto subraya la importancia de una educación equilibrada que fomente la formación de individuos completos y preparados para enfrentar los desafíos de la vida (Gardner, 2011).

5.2. SISTEMAS DE GESTIÓN DE BENEFICIARIOS

Un Sistema de Gestión de Beneficiarios es una plataforma tecnológica diseñada para almacenar, organizar y gestionar la información de los individuos beneficiados por un proyecto social. Estos sistemas permiten un seguimiento más eficiente de los beneficiarios, mejorando la administración de recursos y la programación de actividades en organizaciones con fines sociales (Barrera & Santa Cruz, 2015; Williams, 2017).

5.3. TECNOLOGÍAS WEB

Las tecnologías web abarcan un conjunto de herramientas y técnicas que permiten la creación de aplicaciones accesibles a través de navegadores web. Estas aplicaciones son independientes del dispositivo y del sistema operativo, permitiendo un acceso global y facilitando la interacción entre usuarios y plataformas (Duckett, 2014; Keith, 2013).

5.4. INTERNET

Internet es una red global de dispositivos interconectados que permite la comunicación y el intercambio de información. Desde su creación en la década de 1960, ha evolucionado hasta convertirse en el pilar de la infraestructura tecnológica moderna. Gracias a su arquitectura descentralizada y el uso de protocolos como IP y TCP, permite la transferencia eficiente de datos a escala global (Kurose & Ross, 2020; Tanenbaum & Wetherall, 2011).

5.5. UML

El UML es una notación estándar para el modelado de sistemas de software. Proporciona representaciones gráficas de los componentes, interacciones y comportamientos de un sistema, lo que facilita la comprensión y comunicación entre los equipos de desarrollo. UML se utiliza comúnmente para diseñar, documentar y visualizar sistemas antes de su implementación (Booch, Rumbaugh, & Jacobson, 1999; Fowler, 2004).

5.6. HTML

HTML5 es la versión más actualizada del lenguaje de marcado utilizado para estructurar y presentar contenido en la web. Introduce nuevas etiquetas semánticas, soporte para multimedia (audio, video), gráficos mediante Canvas y SVG, y mejoras en el almacenamiento local, que permiten el desarrollo de aplicaciones web más interactivas y robustas (Duckett, 2014; Keith, 2013).

5.7. CSS3

CSS3 es la evolución del lenguaje de hojas de estilo, utilizado para controlar la apariencia visual de los elementos HTML. Con CSS3, los desarrolladores pueden aplicar diseños flexibles y responsivos a las páginas web, utilizando características avanzadas como transiciones, animaciones y media queries (Keith, 2013; Meyer, 2011).

5.8. JAVASCRIPT

JavaScript es un lenguaje de programación dinámico que se ejecuta en el navegador del cliente, y permite agregar interactividad a las páginas web. A través de la manipulación del DOM y el uso de tecnologías como AJAX y JSON, JavaScript facilita la creación de interfaces de usuario más dinámicas y eficientes (Flanagan, 2020; Crockford, 2008).

5.9. NET CORE 8

.NET Core 8 es un marco de desarrollo de software multiplataforma y de código abierto. Permite la creación de aplicaciones web, móviles y de escritorio escalables y eficientes. En su versión más reciente, .NET Core incluye soporte para contenedores, microservicios y tecnologías avanzadas como inteligencia artificial (Esposito, 2023; Freeman & Sanderson, 2019).

5.10. RAZOR

Razor es un motor de plantillas utilizado en el desarrollo de aplicaciones web con ASP.NET Core. Permite combinar código C# con HTML en archivos únicos, facilitando la generación dinámica de vistas y una mayor flexibilidad en el desarrollo de interfaces de usuario (Freeman, 2018; Steinberg & Lichtenberg, 2018).

5.11. AJAX

AJAX (Asynchronous JavaScript and XML) es una técnica de desarrollo web que permite realizar actualizaciones parciales de la página sin necesidad de recargar por completo. Al facilitar la comunicación asíncrona entre el servidor y el cliente, AJAX mejora la interactividad y la experiencia del usuario en aplicaciones web (Keith, 2011).

5.12. JSON

JSON (JavaScript Object Notation) es un formato ligero de intercambio de datos que se utiliza para transmitir información entre un servidor y una aplicación web. Es ampliamente utilizado en aplicaciones web modernas debido a su simplicidad y su compatibilidad nativa con JavaScript (Cattell, 2018; Crockford, 2006).

5.13. METODOLOGÍAS ÁGILES

Las metodologías ágiles, como Scrum, son enfoques de gestión de proyectos que buscan mejorar la eficiencia y la calidad en el desarrollo de productos, centrándose en la colaboración, la adaptabilidad y la entrega continua de valor (Schwaber & Sutherland, 2020; Beck, 2000; Highsmith, 2002).

Estas metodologías fueron desarrolladas como una respuesta a los enfoques tradicionales (cascada o waterfall), que a menudo resultaban inflexibles ante cambios en los requisitos. En contraste, las metodologías ágiles permiten un proceso iterativo e incremental que facilita la adaptación a nuevas necesidades del cliente o cambios en el entorno del proyecto.

Scrum es uno de los marcos de trabajo más populares dentro de las metodologías ágiles, utilizado especialmente en el desarrollo de software. Su estructura promueve la mejora continua y una retroalimentación constante, garantizando que el producto final esté alineado con las expectativas del cliente o usuario. Scrum divide el trabajo en ciclos cortos y repetitivos denominados sprints, durante los cuales un equipo multidisciplinario desarrolla y entrega incrementos funcionales del producto.

5.14. CONCEPTOS CLAVE DE SCRUM

- (1) **Product Backlog:** Es una lista priorizada de las características, mejoras, y correcciones que se deben implementar en el producto. Este backlog es gestionado por el Product Owner y es flexible, permitiendo que se añadan o modifiquen elementos
- (2) **Sprint Backlog:** Es un subconjunto del Product Backlog que se selecciona al inicio de cada sprint. Representa las tareas que el equipo se compromete a completar durante ese ciclo de trabajo.

- (3) **Sprint:** Es el corazón de Scrum. Un sprint es un ciclo de trabajo de duración fija, generalmente de 2 a 4 semanas, durante el cual se desarrolla una parte funcional del producto. Al finalizar el sprint, se presenta un incremento que debe ser potencialmente entregable.
- (4) **Daily Standup:** Es una breve reunión diaria (máximo 15 minutos) donde el equipo revisa el progreso y ajusta su planificación si es necesario. Cada miembro del equipo responde a tres preguntas clave: ¿Qué hice ayer?, ¿Qué haré hoy?, y ¿Hay algún impedimento?
- (5) **Incremento:** Al final de cada sprint, el equipo produce un incremento del producto que es funcional y puede ser presentado al cliente o stakeholders. Este incremento debe cumplir con la “Definición de Hecho” (Definition of Done), que especifica los criterios de calidad y finalización.
- (6) **Product Owner:** Es el responsable de maximizar el valor del producto, gestionando y asegurando que el equipo esté trabajando en las tareas más prioritarias. Actúa como el enlace entre los stakeholders y el equipo.
- (7) **Scrum Master:** Es quien facilita el proceso de Scrum, asegurándose de que se sigan las reglas y principios del marco. El Scrum Master también trabaja para eliminar los impedimentos que puedan estar bloqueando el progreso del equipo y fomenta un ambiente de trabajo eficiente.
- (8) **Equipo de Desarrollo:** Es un grupo autoorganizado y multidisciplinario que trabaja de manera colaborativa para completar las tareas del Sprint Backlog. No hay jerarquías dentro del equipo de desarrollo, y todos son responsables de entregar el incremento funcional al final de cada sprint.
- (Schwaber & Sutherland, 2020).

5.15. FUNCIONAMIENTO DE SCRUM

Planificación del Sprint (Sprint Planning):

El sprint comienza con una reunión de planificación, donde el equipo de desarrollo, el Product Owner y el Scrum Master se reúnen para definir qué elementos del Product Backlog se incluirán en el Sprint Backlog. Los elementos seleccionados deben ser aquellos que el equipo considera factibles de completar dentro del tiempo establecido.

Ejecución del Sprint:

Durante el sprint, el equipo de desarrollo trabaja en las tareas asignadas, y el Scrum Master asegura que el proceso se desarrolle sin interrupciones. Las reuniones diarias (Daily Standup) se utilizan para sincronizar al equipo y ajustar el plan según sea necesario.

Revisión del Sprint (Sprint Review):

Al final del sprint, el equipo presenta el incremento de trabajo completado al Product Owner y a otros stakeholders en una reunión de revisión. El objetivo es obtener retroalimentación para ajustar el backlog y definir prioridades para el siguiente sprint.

Retrospectiva del Sprint (Sprint Retrospective):

Después de la revisión, el equipo realiza una retrospectiva para evaluar lo que funcionó bien y lo que podría mejorarse en el próximo sprint. Esta retroalimentación ayuda a ajustar el proceso y aumentar la productividad y calidad del equipo en futuros ciclos.

5.16. BENEFICIOS DE SCRUM

- **Flexibilidad:** Scrum permite adaptarse rápidamente a cambios en los requisitos del cliente, priorizando el trabajo más importante.
- **Colaboración:** Promueve la comunicación constante entre los miembros del equipo y con los stakeholders.
- **Entrega Continua:** El enfoque cíclico asegura que se entreguen incrementos del producto regularmente, lo que permite recibir retroalimentación constante.

- Mejora Continua: Las retrospectivas permiten identificar áreas de mejora tanto en el producto como en los procesos del equipo, fomentando un ciclo de optimización continuo.

(Schwaber & Sutherland, 2020).

VI. DISEÑO METODOLÓGICO

El objetivo de aplicar Scrum en el Proyecto Sara María Parrales (**PROYSMAPA**) es mejorar la gestión administrativa y educativa mediante la implementación de un sistema web que optimice los procesos del centro de atención integral. A continuación, se presenta un diseño metodológico basado en Scrum para guiar el desarrollo de este sistema:

6.1. ROLES SCRUM

Product Owner (PO): Representa a los stakeholders y es responsable de definir y priorizar los requisitos del sistema. En el contexto del proyecto Sara María Parrales, este rol puede ser asumido por un coordinador principal que entienda las necesidades de los beneficiarios y el personal del centro.

Scrum Master (SM): Facilita el proceso Scrum, asegura que se sigan las prácticas del marco y elimina impedimentos que puedan afectar el progreso del equipo. Este rol será asumido por uno de los integrantes del grupo.

Equipo de Desarrollo: Un grupo multidisciplinario encargado de diseñar, desarrollar y probar el sistema web. El equipo debe incluir desarrolladores, diseñadores, y expertos en pruebas, preferiblemente con experiencia en tecnologías web y sistemas de gestión.

6.2. ARTEFACTOS SCRUM

Product Backlog: Una lista priorizada de todos los requisitos y características del sistema web. El Product Owner debe mantener y al actualizar el Product Backlog, incluyendo elementos como la gestión de beneficiarios, programación de actividades, reportes administrativos, etc.

Sprint Backlog: Un subconjunto del Product Backlog que el equipo de desarrollo selecciona para trabajar durante un sprint. Cada Sprint Backlog debe incluir las tareas que debe realizar el equipo.

Incremento: La versión funcional del sistema web que se entrega al final de cada sprint. El Incremento debe ser potencialmente desplegable y cumplir con los criterios de aceptación establecidos en el Product Backlog.

6.3. EVENTOS SCRUM

Sprint Planning (Planificación del Sprint): Reunión al inicio de cada sprint donde el Product Owner presenta los elementos del Product Backlog que se van a incluir en el Sprint Backlog. El equipo de desarrollo estima y define las tareas necesarias para completar los elementos seleccionados.

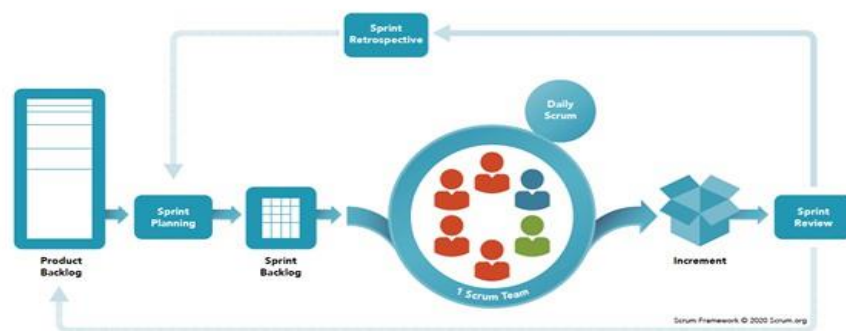
Daily Standup (Reunión Diaria): Reunión breve (15 minutos) que se realiza todos los días del sprint para que el equipo de desarrollo informe sobre el progreso, los obstáculos y los planes para el día. Ayuda a mantener la sincronización y ajustar el trabajo según sea necesario.

Sprint Review (Revisión del Sprint): Reunión al final del sprint donde el equipo presenta el Incremento al Product Owner y a otros stakeholders. Se revisa el trabajo completado y se obtiene retroalimentación para ajustar el Product Backlog y los objetivos del siguiente sprint.

Sprint Retrospective (Retrospectiva del Sprint): Reunión realizada después de la Sprint Review donde el equipo reflexiona sobre el sprint, identifica lo que funcionó bien y lo que se puede mejorar. El objetivo es implementar mejoras en el proceso de futuros sprints.

Figura 1: Metodología SCRUM

Figura 1
Metodología Scrum (Scrum.org, 2023)



Nota: Esta figura muestra la estructura de la metodología scrum.

6.4. SPRINTS PARA PROYSMAPA

Semana 1-2: Sprint 1 – Análisis de Requerimientos y Diseño Inicial

Objetivo: Analizar los requerimientos funcionales y no funcionales del sistema web mediante la recopilación de información con los involucrados claves.

Tareas:

- Planificación de los sprint.
- Definición del Product Backlog inicial.
- Análisis de requisitos con los stakeholders.
- Diseño inicial del sistema, enfocándose en la estructura general.

Detalle del sprint en Anexo 1: Análisis de Requerimientos y Diseño Inicial

Semana 3-4: Sprint 2 – Configuración del Entorno y Arquitectura del Sistema

Objetivo: Diseñar un modelo que cumpla con los requerimientos definidos, utilizando la metodología ágil Scrum y el Lenguaje Unificado de Modelado (UML).

Tareas:

- Desarrollo de la arquitectura del sistema.
- Configuración del entorno de desarrollo.
- Implementación de la base de datos.
- Refinar los diagramas UML basados en el análisis.

Detalle del sprint en Anexo 2: Configuración del Entorno y Arquitectura del Sistema

Semana 5-6: Sprint 3 – Desarrollo Inicial de Módulos e Interfaces

Objetivo: Codificar el sistema web utilizando el framework ASP.NET Core 8 y SQL Server como gestor de bases de datos.

Tareas:

- Desarrollo de módulos iniciales (gestión de usuarios y actividades).
- Creación de interfaces básicas.
- Configuración de herramientas de desarrollo.

Detalle del sprint en Anexo 3: Desarrollo Inicial de Módulos e Interfaces

Semana 7-8: Sprint 4 – Implementación de Funciones Clave (Matricula y Vida Estudiantil).

Objetivo: Implementación de Funcionalidades Clave (Matricula y Vida Estudiantil)

Tarea	Categoría	Asignado a	Tiempo
Desarrollo del módulo de Matrícula (registro y edición)	Módulos académicos	Samuel M.	3h
Desarrollo del módulo de Vida Estudiantil (gestión de historial y eventos)	Módulos académicos	Samuel M.	2.5h
Implementación de lógica de negocio para Matrícula	Lógica de negocio	Nestor A.	1.5h
Implementación de lógica de negocio para Vida Estudiantil	Lógica de negocio	Nestor A.	1.5h
Refactorización de código de formularios y controladores	Refactorización y mantenimiento	Samuel M.	2h
Reorganización de funciones repetidas en métodos reutilizables	Refactorización y mantenimiento	Nestor A.	1.5h
Mejora de legibilidad y limpieza del código (nombres, comentarios, estructura)	Refactorización y mantenimiento	Nestor A.	1.5h
Aplicación de principios SOLID donde corresponde	Buenas prácticas de desarrollo	Samuel M.	2h

Establecimiento de Convenciones de codificación en el equipo	Aseguramiento de calidad	Samuel M.	1h
--	--------------------------	-----------	----

Solución:

Este sprint se centró en la implementación de funcionalidades para el sistema académico. Se integraron los módulos de Matricula y Vida Estudiantil, permitiendo registrar y gestionar elementos esenciales del contexto educativo. Se incorporó la lógica de negocio asociada a cada uno. También se aplicó refactorización de código, con énfasis en mejorar la reutilización y la legibilidad. Este sprint marcó el inicio formal de las buenas prácticas de aseguramiento de calidad.

Semana 9-10: Sprint 5 – Desarrollo de Funcionalidades Adicionales y Revisión del Diseño

Objetivo: Continuar el desarrollo del sistema web, implementando lógica de negocio adicional.

Tareas:

- Desarrollo de funcionalidades adicionales.
- Implementación de la lógica de negocio.
- Revisión y ajustes del diseño según los resultados obtenidos.

Detalle del sprint en Anexo 5: Funcionalidades Adicionales y Revisión del Diseño

Semana 11-12: Sprint 6 – Módulos de Usuario y Seguridad

Objetivo: Desarrollar módulos relacionados con la gestión de usuarios y mejorar la seguridad del sistema.

Tareas:

- Desarrollo de módulos de usuario (autenticación y gestión de perfiles).
- Implementación de medidas de seguridad y autenticación.

- Pruebas de integración para garantizar que los módulos funcionen correctamente.

Detalle del sprint en Anexo 6: Módulos de Usuario y Seguridad

Semana 13-14: Sprint 7 – Finalización de Funcionalidades de Usuario y Pruebas de Usabilidad

Objetivo: Realizar pruebas adicionales y finalizar las funcionalidades relacionadas con la experiencia del usuario.

Tareas:

- Finalización de las funcionalidades de usuario.
- Ajustes y refinamientos en la interfaz.
- Pruebas de usabilidad para asegurar la accesibilidad y facilidad de uso.

Detalle del sprint en Anexo 7: Finalización de Funcionalidades de Usuario y Pruebas de Usabilidad

Semana 15-16: Sprint 8 – Desarrollo de Funcionalidades Avanzadas y Pruebas del Sistema

Objetivo: Desarrollar reportes y funcionalidades avanzadas, además de ejecutar pruebas a nivel de sistema.

Tareas:

- Desarrollo de reportes y funcionalidades avanzadas.
- Implementación de AJAX para mejorar la interacción dinámica.
- Pruebas del sistema para validar la funcionalidad completa.

Detalle del sprint en Anexo 8: Funcionalidades Avanzadas y Pruebas del Sistema

Semana 17-18: Sprint 9 – Implementación de Intercambio de Datos con JSON

Objetivo: Incorporar tecnologías adicionales para mejorar la comunicación y eficiencia del sistema.

Tareas:

- Implementación de JSON para el intercambio eficiente de datos.
- Pruebas de integración final para garantizar la correcta comunicación entre módulos.

Detalle del sprint en Anexo 9: Intercambio de Datos con JSON

Semana 19-20: Sprint 10 – Ajustes Finales y Pruebas de Aceptación

Objetivo: Corregir errores y realizar pruebas de aceptación del usuario para asegurar la calidad del sistema.

Tareas:

- Ajustes finales y corrección de errores detectados.
- Pruebas de aceptación con los usuarios finales.
- Revisión de la documentación del sistema.

Detalle del sprint en Anexo 10: Ajustes Finales y Pruebas de Aceptación

Semana 21-22: Sprint 11 – Preparación para el Despliegue

Objetivo: Preparar el sistema para su despliegue en un entorno de producción.

Tareas:

- Preparación para el despliegue en producción.
- Formación del equipo en el uso y administración del sistema.
- Revisión y ajustes finales basados en la retroalimentación de los usuarios.

Detalle del sprint en Anexo 11: Preparación para el Despliegue

Semana 23-24: Sprint 12 – Despliegue en Producción y Monitoreo

Objetivo: Implementar el sistema en el entorno de producción y asegurar su correcto funcionamiento.

Tareas:

- Despliegue en el entorno de producción.
- Monitoreo inicial y soporte para los usuarios.
- Retroalimentación final y planificación para mejoras futuras.

Detalle del sprint en Anexo 12: Despliegue en Producción y Monitoreo

6.5. PROCESO SCRUM PARA PROYSMAPA

El desarrollo del sistema PROYSMAPA se llevó a cabo siguiendo una secuencia ordenada de fases, materializadas en un conjunto de 12 sprints que, de forma iterativa e incremental, permitieron transformar los requerimientos iniciales en una solución plenamente funcional. El proceso inició con el Sprint 1, enfocado en el análisis de requerimientos funcionales y no funcionales, y la elaboración del diseño inicial. En esta etapa, se definió el Product Backlog, se recopilaron las necesidades clave de los stakeholders y se diseñó la estructura general del sistema, sentando las bases para todo el desarrollo posterior.

En el Sprint 2 se configuró el entorno de desarrollo y se definió la arquitectura del sistema, implementando la base de datos y afinando los diagramas UML de acuerdo con los hallazgos del análisis previo. Esto dio paso al Sprint 3, donde comenzaron a desarrollarse los primeros módulos e interfaces básicas, utilizando ASP.NET Core 8 y SQL Server, con un enfoque en la gestión de usuarios y actividades.

Posteriormente, el Sprint 4 abordó la implementación de funcionalidades clave como Matrícula y Vida Estudiantil, integrando la lógica de negocio respectiva y realizando una refactorización de código para mejorar la legibilidad, la reutilización y la calidad general. Esta etapa marcó el inicio de la adopción sistemática de buenas prácticas de desarrollo y aseguramiento de calidad. En el Sprint 5 se añadieron funcionalidades complementarias, ajustando el diseño según los resultados obtenidos.

En el Sprint 6, la atención se centró en los módulos de usuario y la seguridad del sistema, con la implementación de autenticación y gestión de perfiles, además de medidas adicionales de protección y pruebas de integración. Esto fue complementado en el Sprint 7 con la finalización de funcionalidades de usuario, ajustes en la interfaz y pruebas de usabilidad para garantizar accesibilidad y facilidad de uso.

Las funcionalidades avanzadas y la generación de reportes se desarrollaron en el Sprint 8, donde también se incorporó AJAX para optimizar la interacción dinámica y se realizaron pruebas del sistema. El Sprint 9 integró el intercambio de datos

mediante JSON, asegurando una comunicación eficiente entre módulos y validando la correcta integración global.

En la recta final, el Sprint 10 se dedicó a la corrección de errores y a las pruebas de aceptación con usuarios, verificando la estabilidad y calidad del producto. El Sprint 11 preparó el sistema para su despliegue en producción, incluyendo la capacitación del personal y ajustes finales basados en retroalimentación. Finalmente, el Sprint 12 marcó la puesta en marcha oficial del sistema en el entorno productivo, acompañado de un monitoreo inicial, soporte a usuarios y la planificación de futuras mejoras.

Este diseño metodológico basado en SCRUM proporciono una estructura flexible y colaborativa para el desarrollo del sistema web de PROYSMAPA, optimizando la gestión administrativa y educativa del centro.

VII. ANÁLISIS Y PRESENTACIÓN DE RESULTADOS

7.1. ESTUDIO DE FACTIBILIDAD

El estudio de factibilidad representa una de las fases fundamentales en el desarrollo de cualquier proyecto, ya sea de carácter tecnológico, productivo o de servicios. Su propósito es analizar en profundidad si la propuesta planteada es viable desde diferentes perspectivas, evaluando los recursos disponibles, las capacidades técnicas del equipo, la adaptabilidad del sistema al entorno operativo y la inversión requerida en relación con los beneficios esperados.

En este contexto, el presente capítulo tiene como objetivo valorar si el desarrollo e implementación del sistema web para la gestión educativa del Proyecto Sara María Parrales (PROYSMAPA) es viable en términos operativos, técnicos y económicos. Este análisis permitirá sustentar con claridad y objetividad la ejecución del proyecto y prever sus implicaciones reales en el entorno institucional.

7.2. FACTIBILIDAD OPERATIVA

La factibilidad operativa evalúa si el sistema propuesto puede ser implementado y utilizado eficientemente dentro del contexto organizacional de PROYSMAPA. Este análisis considera factores prácticos, humanos y logísticos que inciden directamente en la funcionalidad del sistema una vez que entre en operación.

En el caso de PROYSMAPA, el sistema web propuesto responde a una necesidad real y urgente de optimizar la gestión de beneficiarios, talleres y actividades administrativas. Su adopción permitiría mejorar la organización interna, reducir errores derivados de la gestión manual y agilizar la toma de decisiones.

No obstante, como en todo proceso de innovación tecnológica, es posible que algunos usuarios manifiesten resistencia o inquietud ante el uso de un nuevo sistema. Para abordar esta situación, se contempla un proceso de capacitación

dirigido a todo el personal que utilizará la plataforma, el cual incluirá sesiones prácticas, materiales de apoyo y un canal de retroalimentación para recibir sugerencias y garantizar una adecuada adaptación.

El perfil del personal que interactúe con el sistema deberá contar con al menos conocimientos básicos en el uso de computadoras. Por su parte, el personal encargado del mantenimiento y soporte del sistema deberá poseer habilidades técnicas específicas, como:

- Experiencia en el uso de tecnologías web.
- Conocimientos en lenguajes como ASP.NET Core 8, JavaScript, Patrón MVC. Manejo de bases de datos relacionales (SQL).
- Capacidad de análisis y resolución de problemas.

El equipo de desarrollo responsable de este proyecto posee los conocimientos necesarios para asumir el mantenimiento del sistema en sus etapas futuras, asegurando así su continuidad operativa.

7.3. FACTIBILIDAD TÉCNICA

La factibilidad técnica se centra en determinar si existen los recursos tecnológicos y las capacidades técnicas necesarias para desarrollar e implementar el sistema de manera eficiente. Esta evaluación considera el acceso a herramientas de desarrollo, lenguajes de programación, bases de datos, y plataformas de despliegue, entre otros factores clave.

El sistema web fue desarrollado utilizando ASP.NET Core 8 como framework principal, lo cual garantiza estabilidad, escalabilidad y un rendimiento óptimo en entornos web. Para la gestión de la base de datos se utilizará SQL Server, una tecnología robusta y ampliamente utilizada en entornos corporativos.

Además, se prevé el uso de servicios en la nube o servidores locales para el despliegue de la aplicación, según las capacidades de infraestructura del proyecto.

Las interfaces del sistema están diseñadas para ser accesibles desde navegadores web, asegurando una experiencia amigable para los usuarios.

El equipo cuenta con las herramientas necesarias para la construcción del sistema, incluyendo:

Tabla 1: Herramientas para el sistema

Tipo de Recurso	Aplicación	Recurso
Software	Análisis del sistema	Miro.com – Versión Free
Software	Desarrollo	Visual Studio 2022
Conocimiento	Diseño y Lenguajes de Programación	C# JavaScript HTML / CSS Asp .Net Core 8
Software	Documentación	Office Libre
Software	Sistema Operativo	Windows 10 o superior
Hardware	Dispositivos Clientes PC	Intel Pentium Dual Core o Superior 4 GB RAM o Superior Monitor 21'' o superior con una resolución de 1024 x 768
Hardware	Servidor	Procesador a 2.5 GHz o superior 16 GB RAM o Superior
Hardware	Desarrollo	Intel Core i5 12100 16 GB RAM Monitor 27'' con una resolución de 1024 x 768

Nota. Elaboración propia.

7.4. FACTIBILIDAD ECONÓMICA

La factibilidad económica analiza la relación entre los costos de desarrollo, implementación y mantenimiento del sistema, frente a los beneficios que este generará a corto, mediano y largo plazo. En este caso, dado que el sistema será desarrollado como parte de un trabajo monográfico, no se incurre en costos directos por concepto de contratación de servicios de desarrollo o licencias comerciales.

Los recursos necesarios para el proyecto son gestionables dentro de los medios actuales del equipo y la organización:

El equipo de desarrollo ha estimado el costo de los recursos humanos tomando como referencia sondeos realizados a empresas privadas de desarrollo de software en el país. Esta estimación se basa en un esquema de contratación temporal para el período de desarrollo del sistema.

A continuación, se detalla el desglose del recurso humano considerado:

Tabla 2: Costo de los desarrolladores

Recurso	Cantidad	Costo Mes	Duración	Total
Desarrollador	2	800	5 meses	\$8,000.00

Nota. Elaboración propia.

Además, se contempla la capacitación del personal que utilizará el sistema a través de talleres internos organizados por el equipo de desarrollo, por lo cual no será necesario incurrir en gastos adicionales para la contratación de instructores externos.

También se considera el uso de los equipos de cómputo personales del equipo desarrollador, por lo que no se requiere inversión adicional en hardware para el desarrollo del sistema. Todos los costos relevantes están contemplados en la tabla anterior, no previéndose gastos adicionales fuera de los ya estimados.

Para la implementación del sistema web propuesto, se ha considerado el uso de un servidor local como alternativa económica y funcional, especialmente adecuada para el contexto operativo de PROYSMAPA. Esta decisión busca garantizar el control interno de la información, evitar gastos recurrentes en servicios en la nube y facilitar el acceso al sistema dentro de las instalaciones del proyecto.

El servidor tendrá como función principal alojar la aplicación web, gestionar la base de datos, y permitir el acceso a los usuarios autorizados de forma segura dentro de la red local del centro. Dado que se trata de una solución de bajo volumen de usuarios y tráfico moderado, se ha estimado que un servidor con características básicas será suficiente para garantizar un rendimiento adecuado.

A continuación, se presenta un presupuesto estimado para la adquisición de un servidor Windows básico:

Tabla 3: Requisitos para Servidor Web

Componente	Especificaciones mínimas sugeridas	Costo aproximado
CPU	Intel Core i5 (10ª gen o superior)	\$250.00
Memoria RAM	16 GB DDR4	\$60.00
Almacenamiento	SSD 512 GB	\$60.00
Placa base	Compatible con procesador y RAM seleccionados	\$100.00
Fuente de poder	500W	\$40.00
Gabinete	Estándar ATX	\$40.00
Sistema operativo	Licensee Windows Server 2022 Standard (OEM)	\$500.00
Estabilizador o UPS	Básico de 1000VA	\$80.00
Monitor, teclado, mouse	Periféricos básicos	\$80.00
Total		\$1,210.00

Nota. Elaboración propia.

Este servidor cubriría adecuadamente las necesidades del sistema web desarrollado, permitiendo su implementación y operación sin dependencia de servicios externos. En caso de crecimiento futuro o necesidad de acceso remoto, se podrá escalar la solución o migrarla hacia un entorno en la nube.

Finalmente, se presenta un resumen general de los costos asociados al desarrollo e implementación del sistema web propuesto.

Tabla 4: Costo Total del Proyecto

Recurso	Costo
Costo de Desarrollo	\$ 8,000.00
Costo de Servidor	\$ 1,210.00
Total, Estimado	\$ 9,210.00

Nota. Elaboración propia.

Este análisis integral permite visualizar de manera clara y ordenada la inversión estimada requerida para llevar a cabo el proyecto, proporcionando así una base sólida para la toma de decisiones y la planificación financiera del mismo.

7.5. REQUERIMIENTOS DEL SISTEMA

El levantamiento de requerimientos representa una etapa fundamental para el desarrollo del sistema web de gestión administrativa y educativa de PROYSMAPA. Dado el contexto comunitaria y educativa del proyecto, esta fase se llevó a cabo con un enfoque participativo, buscando no solo la eficiencia técnica, sino también el respeto por la misión y los valores del centro.

Para definir los objetivos y el alcance inicial del proyecto, se realizaron entrevistas estructuradas con los stakeholders y usuarios clave, incluyendo a miembros del equipo administrativo, docentes voluntarios y otros colaboradores. Estas sesiones de trabajo permitieron identificar las experiencias, desafíos y aspiraciones del personal en relación con el manejo de la información y la gestión educativa. Las preguntas formuladas se orientaron a identificar las necesidades reales del sistema. Las preguntas usadas en las entrevistas se encuentran en Anexo 13: Entrevista a PROYSMAPA

Como resultado de este proceso de análisis, se obtuvieron las historias de usuario que reflejaban las situaciones reales del día a día del centro. Este análisis permitió estructurar la primera fase de desarrollo, la cual se materializó en el Sprint 1, que se llevó a cabo durante las semanas 1 y 2 del proceso Scrum. Durante este sprint, se tradujeron los requerimientos identificados en módulos funcionales concretos que permiten una visión estructurada y escalable del sistema.

Los requerimientos se clasificaron en dos grandes grupos:

Requerimientos funcionales: Son aquellos que describen las funciones específicas que el sistema debe cumplir, como el registro de beneficiarios, control de asistencia, planificación de actividades, y generación de reportes.

Requerimientos no funcionales: Son aquellos que definen las características de calidad del sistema, como la seguridad, usabilidad, rendimiento, y disponibilidad dentro de un entorno local.

Nota: Las siguientes tablas son elaboración propia.

Tabla 5: Requerimientos Profesiones

Profesiones	
RF 01	El sistema deberá permitir el registro de nuevas profesiones asociadas a las personas que colaboran o colaborarán en la fundación.
RF 02	El sistema validará que cada nombre de profesión sea único, evitando duplicidades en el registro.
RF 03	El sistema deberá permitir la asociación de una profesión con una o más personas registradas en la base de datos.
RF 04	El sistema permitirá la visualización en formato de lista de todas las profesiones activas almacenadas.
RF 05	El sistema deberá permitir la edición de los datos de una profesión previamente registrada.
RF 06	El sistema mostrará la opción de dar de baja profesiones que ya no se utilicen, siempre que no estén vinculadas a personas activas.
RF 07	El sistema deberá permitir la exportación de la información de profesiones en formatos compatibles

Tabla 6: Requerimientos Aulas

Aulas	
RF 08	El sistema deberá permitir el registro de aulas físicas o espacios de trabajo utilizados por la fundación.
RF 09	El sistema validará que el nombre de cada aula sea único, evitando duplicidades en el registro.
RF 10	El sistema deberá permitir la asociación de un aula con uno o más cursos registrados en la base de datos.
RF 11	El sistema permitirá la visualización en formato de lista de todas las aulas activas registradas.
RF 12	El sistema deberá permitir la edición de los datos de un aula previamente registrada.
RF 13	El sistema mostrará la opción de dar de baja aquellas aulas que ya no se utilicen, siempre que no estén vinculadas a cursos activos.
RF 14	El sistema deberá permitir la exportación de la información de las aulas en formatos compatibles.

Tabla 7: Requerimiento Ciclo Académico

Ciclo Académico	
RF 15	El sistema deberá permitir el registro de nuevos ciclos académicos planificados por la fundación.
RF 16	El sistema deberá validar que el nombre de cada ciclo académico sea único, evitando duplicidades y garantizando la integridad de los datos.
RF 17	El sistema deberá permitir la asociación de un ciclo académico con uno o más alumnos, aulas, matrículas y otros registros administrativos necesarios para su correcta gestión.
RF 18	El sistema permitirá la visualización, en formato de lista, de todos los ciclos académicos activos registrados en la base de datos, incluyendo información relevante como estado y duración.
RF 19	El sistema deberá permitir la edición de los datos de un ciclo académico previamente registrado, siempre que no se haya cerrado oficialmente.
RF 20	El sistema deberá permitir la baja lógica (desactivación) de ciclos académicos que ya no estén en uso, siempre y cuando no estén asociados a cursos activos o en ejecución.

RF 21	El sistema deberá permitir la exportación de la información de los ciclos académicos en formatos compatibles.
--------------	---

Tabla 8: Requerimientos Cursos

Cursos	
RF 22	El sistema deberá permitir el registro de cursos académicos ofrecidos por la fundación,
RF 23	El sistema deberá validar que el nombre de cada curso sea único, evitando duplicidades en el registro.
RF 24	El sistema deberá permitir la asociación de un curso con uno o más estudiantes registrados en la base de datos, permitiendo así su inscripción.
RF 25	El sistema permitirá la visualización en formato de lista de todos los cursos activos registrados, con opciones de búsqueda y filtrado.
RF 26	El sistema deberá permitir la edición de los datos de un curso previamente registrado, siempre que el curso no esté cerrado o finalizado.
RF 27	El sistema mostrará la opción de dar de baja aquellos cursos que ya no se utilicen, siempre y cuando no estén vinculados a ciclos académicos activos o estudiantes inscritos.
RF 28	El sistema deberá permitir la exportación de la información de los cursos en formatos compatibles.

Tabla 9: Requerimientos Turnos

Turnos	
RF 29	El sistema deberá permitir el registro de turnos ofrecidos por la fundación.
RF 30	El sistema validará que el nombre de cada turno sea único, evitando duplicidades en el registro y asegurando una identificación clara.
RF 31	El sistema deberá permitir la asociación de un turno con uno o más cursos registrados en la base de datos, facilitando así la gestión de inscripciones y horarios.
RF 32	El sistema permitirá la visualización en formato de lista de todos los turnos activos registrados, incluyendo opciones de búsqueda, filtrado y ordenamiento por criterios relevantes.
RF 33	El sistema deberá permitir la edición de los datos de un turno previamente registrado, siempre que el turno no se encuentre en estado cerrado o finalizado.
RF 34	El sistema mostrará la opción de dar de baja aquellos turnos que ya no se utilicen, siempre que no estén vinculados a ciclos académicos activos o estudiantes inscritos.
RF 35	El sistema deberá permitir la exportación de la información de los turnos en formatos compatibles.

Tabla 10: Requerimientos Matricula

Matriculas	
RF 35	El sistema deberá permitir el registro de matrículas para estudiantes de la fundación.
RF 36	El sistema validará que cada matrícula sea única en relación con el curso, turno, ciclo académico y estudiante, evitando duplicidades.
RF 37	El sistema deberá permitir la asociación de una matrícula con un curso, estudiante, ciclo académico y turno previamente registrados en la base de datos.
RF 38	El sistema permitirá la visualización en formato de lista de todas las matrículas registradas, incluyendo funcionalidades de búsqueda, filtrado y ordenamiento por criterios relevantes como nombre del estudiante, curso o ciclo académico.
RF 39	El sistema deberá permitir la edición de los datos de una matrícula previamente registrada, siempre que el turno correspondiente no esté cerrado o finalizado.

RF 40	El sistema mostrará la opción de dar de baja aquellas matrículas que ya no estén en uso o que deban ser anuladas por razones administrativas o académicas.
RF 41	El sistema deberá permitir la exportación de la información de las matrículas en formatos compatibles.

Tabla 11: Requerimientos Personas

Personas	
RF 42	El sistema deberá permitir el registro de personas naturales, tanto dentro como fuera de la fundación.
RF 43	El sistema deberá permitir clasificar a las personas registradas como beneficiarios, estudiantes o trabajadores, con el fin de gestionar de manera diferenciada cada tipo de registro.
RF 44	El sistema validará que cada persona registrada sea única mediante el uso de su documento de identidad correspondiente, evitando duplicidades en el registro.
RF 45	El sistema deberá permitir la asignación de roles o categorías a cada persona, pudiendo ser registrada como estudiante o trabajador, según corresponda.
RF 46	El sistema permitirá la visualización en formato de lista de todas las personas registradas, con opciones de búsqueda, filtrado y ordenamiento según diversos criterios
RF 47	El sistema deberá permitir la edición de los datos de una persona previamente registrada, siempre y cuando esta esté activa en el sistema.
RF 48	El sistema mostrará la opción de dar de baja a aquellas personas que ya no serán relacionadas al proyecto, eliminando o inactivando su registro de acuerdo con las necesidades administrativas.
RF 49	El sistema deberá permitir la exportación de la información de las personas registradas en formatos compatibles.

Tabla 12: Requerimientos Tipo de Identificación

Tipo de Identificación	
RF 50	El sistema deberá permitir el registro de tipos de identificación para personas naturales, tanto internas como externas a la fundación.
RF 51	El sistema validará que cada tipo de identificación registrado sea único, evitando duplicidades.
RF 52	El sistema deberá permitir la asociación de un tipo de identificación con su respectivo número asignado a cada persona registrada.
RF 53	El sistema permitirá la visualización en formato de lista de todos los tipos de identificación registrados, incluyendo funcionalidades de búsqueda, filtrado y ordenamiento por criterios relevantes.
RF 54	El sistema deberá permitir la edición de los datos de un tipo de identificación previamente registrado, siempre que no esté vinculado a procesos cerrados o finalizados.
RF 55	El sistema mostrará la opción de dar de baja aquellos tipos de identificación que ya no estén en uso o que deban ser anulados por motivos administrativos o académicos.
RF 56	El sistema deberá permitir la exportación de la información de los tipos de identificación en formatos compatibles.

Tabla 13: Requerimientos Asistencia Escolar

Asistencia Escolar	
RF 57	El sistema deberá permitir el registro de la asistencia diaria de los estudiantes a los cursos en los que están inscritos.

RF 58	El sistema deberá validar que la asistencia se registre de forma simplificada por día, asegurando que no se dupliquen registros para una misma fecha, estudiante y curso.
RF 59	El sistema deberá permitir la visualización en formato de lista de los estudiantes según los filtros aplicados (curso, ciclo académico, fecha, entre otros) para facilitar la marcación de asistencia.
RF 60	El sistema deberá permitir la edición de los registros de asistencia, en caso de que se detecten errores o ajustes posteriores.
RF 61	El sistema deberá permitir la exportación de los registros de asistencia en formatos compatibles.

Tabla 14: Requerimientos Notas Escolares

Notas Escolares	
RF 62	El sistema deberá permitir el registro de las notas académicas de los estudiantes inscritos en un curso, distribuidas en cuatro evaluaciones con un valor del 25% cada una.
RF 63	El sistema deberá validar que las notas se registren de forma simplificada y precisa, evitando duplicidad de registros para un mismo estudiante en un curso determinado.
RF 64	El sistema deberá permitir la visualización de los registros de notas en formato de lista, aplicando filtros como curso, ciclo académico, estudiante y fecha.
RF 65	El sistema deberá permitir la edición de los registros de notas existentes, siempre que se requieran ajustes por correcciones o cambios académicos.
RF 66	El sistema deberá permitir la exportación de los registros de notas escolares en formatos compatibles.

Tabla 15: Requerimientos Consulta Médicas y Psicológicas

Consultas Médicas y Psicológicas	
RF 67	El sistema deberá permitir el registro de consultas médicas y psicológicas ofrecidas por la fundación.
RF 68	El sistema validará que el nombre de cada consulta sea único, evitando duplicidades en el registro y asegurando que los datos ingresados sean correctos.
RF 69	El sistema deberá permitir la asociación de una consulta con un estudiante, beneficiario o trabajador que la requiera.
RF 70	El sistema permitirá la visualización en formato de lista de todas las consultas registradas, incluyendo opciones de búsqueda, filtrado y ordenamiento por criterios relevantes.
RF 71	El sistema deberá permitir la edición de los datos de una consulta cuando se requieran ajustes o correcciones.
RF 72	El sistema deberá permitir la exportación de la información de las consultas en formatos compatibles como PDF o Excel.

7.6. CASOS DE USO DEL SISTEMA

Este capítulo presenta los casos de uso del sistema propuesto, los cuales fueron elaborados a partir de los requerimientos funcionales definidos durante el Sprint 1 y refinados en el Sprint 2 del proceso de desarrollo. En este segundo sprint, correspondiente a las semanas 3 y 4, se configuró el entorno de desarrollo y se diseñó la arquitectura del sistema con base en los lineamientos establecidos.

Durante este sprint se generaron y refinaron los diagramas de casos de uso, los cuales describen de forma estructurada cómo interactúan los actores con el sistema para cumplir tareas específicas. Estos diagramas fueron fundamentales para guiar el diseño técnico y funcional del sistema, y para asegurar que las funcionalidades respondieran a las necesidades identificadas por la Fundación Sara María Parrales (PROYSMAPA).

Nota. Las siguientes tablas de casos de uso (CU) son elaboración propia.

Tabla 16: Registrar Profesión

Elemento	Descripción
Código	CU-01
Nombre	Registrar profesión
Actor principal	Administrador
Descripción	El sistema permite al administrador registrar una nueva profesión.
Precondiciones	El usuario debe estar autenticado como administrador.
Flujo principal	El administrador accede al módulo de profesiones. Selecciona la opción “Registrar nueva profesión”. Ingresa el nombre de la profesión. El sistema valida que el nombre no exista previamente. El sistema guarda la nueva profesión y confirma el registro.
Postcondiciones	La nueva profesión queda almacenada en la base de datos.
Reglas de negocio	No se permiten nombres duplicados.

Tabla 17: Editar Profesión

Elemento	Descripción
Código	CU-02
Nombre	Editar profesión
Actor principal	Administrador
Descripción	El sistema permite modificar los datos de una profesión existente.
Precondiciones	Debe existir al menos una profesión registrada.
Flujo principal	El administrador accede a la lista de profesiones. Selecciona la opción “Editar”. Modifica los datos necesarios. El sistema guarda los cambios y notifica la edición exitosa.
Postcondiciones	La información actualizada queda reflejada en el sistema.

Tabla 18: Dar baja profesión

Elemento	Descripción
Código	CU-04
Nombre	Dar de baja una profesión
Actor principal	Administrador
Descripción	El sistema permite dar de baja una profesión que no esté en uso.
Precondiciones	La profesión no debe estar asociada a personas activas.
Flujo principal	El administrador accede a la lista de profesiones. Selecciona una profesión sin vínculos. El sistema cambia el estado o elimina el registro. Se notifica la acción.
Postcondiciones	La profesión queda inactiva o eliminada del sistema.
Reglas de negocio	No se permite eliminar profesiones vinculadas a personas activas.

Tabla 19: Asociar profesión a personas

Elemento	Descripción
Código	CU-05
Nombre	Asociar profesión a persona
Actor principal	Administrador
Descripción	El sistema permite asociar una o más profesiones a personas registradas.
Precondiciones	Deben existir personas y profesiones activas.
Flujo principal	El administrador accede al perfil de una persona. Selecciona las profesiones. El sistema guarda la relación.
Postcondiciones	La persona queda asociada a una o más profesiones.
Reglas de negocio	Solo se pueden asociar profesiones activas.

Tabla 20: Validar unicidad de la profesión

Elemento	Descripción
Código	CU-06
Nombre	Validar unicidad del nombre de profesión
Actor principal	Sistema (automático)
Descripción	El sistema verifica que el nombre de la profesión no esté duplicado.
Precondiciones	El administrador está registrando o editando una profesión.
Flujo principal	El sistema detecta la entrada del nombre. Busca en la base de datos. Si existe, muestra error; si no, permite continuar.
Postcondiciones	El proceso continúa o se detiene según validación.
Reglas de negocio	No se permiten nombres de profesión duplicados.

Tabla 21: Exportar lista de profesiones

Elemento	Descripción
Código	CU-07
Nombre	Exportar lista de profesiones
Actor principal	Administrador
Descripción	El sistema permite exportar la información de profesiones activas.
Precondiciones	Existen datos de profesiones disponibles.

Flujo principal	El administrador accede al módulo de profesiones. Selecciona "Exportar". El sistema genera y descarga el archivo.
Postcondiciones	Archivo descargado con la información solicitada.
Reglas de negocio	La exportación incluye únicamente profesiones activas.

Tabla 22: Registrar aula

Elemento	Descripción
Código	CU-08
Nombre	Registrar aula
Actor principal	Administrador
Descripción	Permite al administrador registrar una nueva aula o espacio físico de trabajo en la fundación.
Precondiciones	El usuario debe estar autenticado con permisos de administrador.
Flujo principal	El administrador accede al módulo de aulas. Selecciona la opción "Registrar aula". Ingresa el nombre del aula. El sistema valida que el nombre no exista. Se guarda la nueva aula y se notifica el registro exitoso.
Postcondiciones	La nueva aula queda registrada como activa en la base de datos.
Reglas de negocio	No se permiten nombres duplicados de aula.

Tabla 23: Validar nombre único de aula

Elemento	Descripción
Código	CU-09
Nombre	Validar nombre único de aula
Actor principal	Sistema (automático)
Descripción	El sistema valida que el nombre del aula no exista previamente para evitar duplicidades.
Precondiciones	El administrador debe estar registrando o editando una aula.
Flujo principal	El administrador introduce el nombre de aula. El sistema consulta la base de datos. Si el nombre existe, muestra un mensaje de advertencia. Si no existe, permite continuar con el proceso.
Postcondiciones	El proceso de registro o edición continúa si el nombre es único.
Reglas de negocio	No se permite registrar dos aulas con el mismo nombre.

Tabla 24: Visualizar aulas activas

Elemento	Descripción
Código	CU-10
Nombre	Visualizar aulas activas
Actor principal	Administrador
Descripción	El sistema muestra en formato de lista todas las aulas activas disponibles en la base de datos.
Precondiciones	Debe haber al menos una aula registrada y activa.

Flujo principal	El administrador accede al módulo de aulas. El sistema carga y muestra la lista de aulas activas con su información relevante.
Postcondiciones	Las aulas activas son visualizadas correctamente.
Reglas de negocio	Solo se listan aulas con estado activo.

Tabla 25: Editar datos de un aula

Elemento	Descripción
Código	CU-11
Nombre	Editar datos de un aula
Actor principal	Administrador
Descripción	Permite modificar la información de un aula previamente registrada.
Precondiciones	El aula debe estar registrada en el sistema.
Flujo principal	El administrador accede a la lista de aulas. Selecciona la opción "Editar" en un aula. Realiza los cambios necesarios. El sistema guarda los cambios y confirma la actualización.
Postcondiciones	Los cambios realizados se reflejan en el sistema.
Reglas de negocio	El sistema no permite nombres duplicados al editar.

Tabla 26: Dar de baja un aula

Elemento	Descripción
Código	CU-12
Nombre	Dar de baja un aula
Actor principal	Administrador
Descripción	Permite dar de baja un aula que ya no esté en uso y que no esté vinculada a cursos activos.
Precondiciones	El aula no debe tener vínculos activos.
Flujo principal	El administrador accede a la lista de aulas. Selecciona el aula. El sistema verifica que no haya relaciones activas. Cambia el estado del aula a inactiva o la elimina (según configuración).
Postcondiciones	El aula queda inactiva o eliminada.
Reglas de negocio	No se puede dar de baja un aula asociada a cursos activos.

Tabla 27: Exportar información de aulas

Elemento	Descripción
Código	CU-13
Nombre	Exportar información de aulas
Actor principal	Administrador
Descripción	Permite exportar los datos de las aulas a formatos compatibles como PDF o Excel.
Precondiciones	Deben existir aulas registradas en el sistema.
Flujo principal	El administrador accede al listado de aulas. Selecciona la opción "Exportar". El sistema genera el archivo y permite su descarga.
Postcondiciones	El archivo exportado queda disponible para su uso.

Reglas de negocio	Solo se exportan aulas activas si así está configurado.
--------------------------	---

Tabla 28: Registrar ciclo académico

Elemento	Descripción
Código	CU-14
Nombre	Registrar ciclo académico
Actor principal	Administrador
Descripción	Permite registrar nuevos ciclos académicos con su información correspondiente.
Precondiciones	El administrador debe estar autenticado en el sistema.
Flujo principal	El administrador accede al módulo de ciclos académicos. Selecciona la opción "Registrar nuevo ciclo". Completa el formulario con los datos del ciclo académico. El sistema valida la información. El sistema guarda el nuevo ciclo en la base de datos. Se muestra mensaje de confirmación.
Postcondiciones	El ciclo académico queda registrado correctamente.
Reglas de negocio	El nombre del ciclo debe ser único.

Tabla 29: Validar nombre único del ciclo académico

Elemento	Descripción
Código	CU-15
Nombre	Validar nombre único del ciclo académico
Actor principal	Sistema
Descripción	Verifica que el nombre del ciclo académico no se repita en el sistema.
Precondiciones	El administrador debe haber ingresado un nombre para el ciclo.
Flujo principal	El administrador ingresa el nombre del ciclo. El sistema consulta la base de datos. Si el nombre ya existe, muestra mensaje de error. Si el nombre es único, permite continuar con el registro.
Postcondiciones	El registro continúa si el nombre es válido.
Reglas de negocio	No se permiten ciclos con nombres duplicados.

Tabla 30: Visualizar lista de ciclos académicos

Elemento	Descripción
Código	CU-16
Nombre	Visualizar lista de ciclos académicos
Actor principal	Administrador
Descripción	Muestra una lista de los ciclos académicos activos en el sistema.
Precondiciones	Debe existir al menos un ciclo registrado.
Flujo principal	El administrador accede al módulo de ciclos académicos. El sistema carga y muestra la lista de ciclos activos. El administrador puede consultar detalles individuales si lo desea.
Postcondiciones	Los ciclos académicos se visualizan correctamente.

Reglas de negocio	Se visualizan solo ciclos activos.
--------------------------	------------------------------------

Tabla 31: Editar ciclo académico

Elemento	Descripción
Código	CU-17
Nombre	Editar ciclo académico
Actor principal	Administrador
Descripción	Permite modificar los datos de un ciclo académico previamente registrado.
Precondiciones	El ciclo académico debe estar activo.
Flujo principal	El administrador accede al listado de ciclos. Selecciona el ciclo a editar. Modifica los campos necesarios. El sistema valida y guarda los cambios. Se muestra un mensaje de confirmación.
Postcondiciones	Los datos del ciclo académico son actualizados.
Reglas de negocio	No se permite duplicidad en el nombre tras la edición.

Tabla 32: Dar de baja ciclo académico

Elemento	Descripción
Código	CU-18
Nombre	Dar de baja ciclo académico
Actor principal	Administrador
Descripción	Permite desactivar un ciclo académico que ya no se utilizará, siempre que no esté vinculado a cursos activos.
Precondiciones	El ciclo académico no debe tener dependencias activas.
Flujo principal	El administrador accede al listado de ciclos. Selecciona el ciclo a dar de baja. El sistema verifica que no esté vinculado a cursos activos. El administrador confirma la baja. El sistema desactiva el ciclo académico.
Postcondiciones	El ciclo académico queda con estado inactivo.

Tabla 33: Nombre Exportar información de ciclos académicos

Elemento	Descripción
Código	CU-19
Nombre	Exportar información de ciclos académicos
Actor principal	Administrador
Descripción	Permite exportar los datos de los ciclos académicos en formatos compatibles como Excel.
Precondiciones	Deben existir ciclos académicos registrados.
Flujo principal	El administrador accede al listado de ciclos académicos. Selecciona la opción "Exportar". El sistema genera el archivo. El administrador descarga el archivo.
Postcondiciones	El archivo exportado queda disponible para su uso.

s	
Reglas de negocio	Solo se exportan ciclos permitidos por los filtros del sistema.

Tabla 34: Registrar curso académico

Elemento	Descripción
Código	CU-20
Nombre	Registrar curso académico
Actor principal	Administrador
Descripción	Permite registrar un nuevo curso académico en el sistema.
Precondiciones	El usuario debe tener permisos de administrador.
Flujo principal	El administrador accede al módulo de administración de cursos. Selecciona la opción "Registrar nuevo curso". Completa el formulario con los datos requeridos. El sistema valida que el nombre del curso no esté repetido. El sistema guarda el nuevo curso y muestra un mensaje de confirmación.
Postcondiciones	El curso queda registrado en el sistema.
Reglas de negocio	El nombre del curso debe ser único.

Tabla 35: Validar nombre único de curso

Elemento	Descripción
Código	CU-21
Nombre	Validar nombre único de curso
Actor principal	Sistema
Descripción	Verifica que el nombre del curso ingresado no exista previamente.
Precondiciones	El administrador debe estar registrando o editando un curso.
Flujo principal	El administrador ingresa el nombre del curso. El sistema consulta si ya existe un curso con ese nombre. Si existe, muestra un mensaje de error. Si no existe, permite continuar con el registro o edición.
Postcondiciones	El administrador puede continuar si el nombre es válido.

Tabla 36: Visualizar lista de cursos activos

Elemento	Descripción
Código	CU-22
Nombre	Visualizar lista de cursos activos
Actor principal	Administrador
Descripción	Permite consultar todos los cursos activos en el sistema.
Precondiciones	Debe haber cursos registrados.
Flujo principal	El administrador accede al módulo de cursos. El sistema muestra la lista de cursos activos. 3. El administrador puede aplicar filtros y búsquedas según criterios (nombre, estado, duración).
Postcondiciones	El administrador visualiza correctamente los cursos activos.
Reglas de negocio	Solo se muestran cursos con estado activo.

Tabla 37: Editar información de curso

Elemento	Descripción
Código	CU-23
Nombre	Editar información de curso
Actor principal	Administrador
Descripción	Permite modificar los datos de un curso previamente registrado.
Precondiciones	El curso debe estar activo y editable.
Flujo principal	El administrador accede a la información de un curso. Selecciona la opción "Editar". Modifica los campos deseados. El sistema guarda los cambios y muestra una confirmación.
Postcondiciones	El curso se actualiza con los nuevos datos.
Reglas de negocio	No se puede cambiar el nombre a uno ya existente.

Tabla 38: Dar de baja curso

Elemento	Descripción
Código	CU-24
Nombre	Dar de baja curso
Actor principal	Administrador
Descripción	Permite desactivar un curso que ya no será utilizado.
Precondiciones	El curso no debe estar vinculado a un ciclo académico activo o a estudiantes inscritos.
Flujo principal	El administrador accede a la información de un curso. Selecciona la opción "Dar de baja". El sistema valida las condiciones de desvinculación. Si cumple, el sistema desactiva el curso y notifica la acción.
Postcondiciones	El curso queda inactivo y no podrá ser asignado.
Reglas de negocio	No se pueden dar de baja cursos con dependencias activas.

Tabla 39: Exportar información de cursos

Elemento	Descripción
Código	CU-25
Nombre	Exportar información de cursos
Actor principal	Administrador
Descripción	Permite exportar la información de los cursos en formatos como PDF o Excel.
Precondiciones	Deben existir cursos registrados en el sistema.
Flujo principal	El administrador accede a la lista de cursos. Selecciona la opción "Exportar". El sistema genera el archivo y permite su descarga.
Postcondiciones	El archivo queda disponible para el administrador.
Reglas de negocio	Solo se exportan cursos según los filtros aplicados.

Tabla 40: Registrar turno

Elemento	Descripción
Código	CU-26
Nombre	Registrar turno
Actor principal	Administrador
Descripción	Permite al administrador registrar un nuevo turno ofrecido por la fundación.
Precondiciones	El administrador debe estar autenticado en el sistema.
Flujo principal	El administrador accede al módulo de turnos. Selecciona la opción "Registrar nuevo turno". Ingresa la información requerida. El sistema valida que el nombre no esté repetido. El sistema guarda el nuevo turno y muestra mensaje de éxito.
Postcondiciones	El turno queda registrado y disponible en el sistema.
Reglas de negocio	El nombre del turno debe ser único.

Tabla 41: Validar nombre de turno único

Elemento	Descripción
Código	CU-27
Nombre	Validar nombre de turno único
Actor principal	Sistema
Descripción	Garantiza que cada turno registrado tenga un nombre único.
Precondiciones	Se está intentando registrar o editar un turno.
Flujo principal	El sistema verifica el nombre ingresado contra los nombres existentes. Si el nombre ya existe, muestra un mensaje de error. Si el nombre es único, permite continuar con el registro o edición.
Postcondiciones	Se valida la unicidad del nombre antes de guardar.
Reglas de negocio	No se permite duplicidad en los nombres de turnos.

Tabla 42: Visualizar lista de turnos activos

Elemento	Descripción
Código	CU-28
Nombre	Visualizar lista de turnos activos
Actor principal	Administrador
Descripción	Muestra todos los turnos activos registrados en el sistema con opciones de búsqueda y filtrado.
Precondiciones	Existen turnos registrados.
Flujo principal	El administrador accede al módulo de turnos. Visualiza una lista con los turnos activos. El sistema muestra los resultados filtrados.
Postcondiciones	Se despliega correctamente la información de los turnos activos.
Reglas de negocio	Solo se visualizan turnos con estado activo.

Tabla 43: Editar turno

Elemento	Descripción
Código	CU-29
Nombre	Editar turno
Actor principal	Administrador
Descripción	Permite modificar la información de un turno siempre que este no esté cerrado o finalizado.
Precondiciones	El turno debe estar en estado editable.
Flujo principal	El administrador accede al detalle del turno. Realiza los cambios necesarios. Guarda la información. El sistema valida y actualiza el turno.
Postcondiciones	La información del turno se actualiza correctamente.
Reglas de negocio	No se puede editar un turno cerrado o finalizado.

Tabla 44: Dar de baja turno

Elemento	Descripción
Código	CU-30
Nombre	Dar de baja turno
Actor principal	Administrador
Descripción	Permite desactivar un turno si no está vinculado a ciclos activos ni estudiantes inscritos.
Precondiciones	El turno debe estar libre de vinculaciones activas.
Flujo principal	El administrador accede al listado de turnos. Selecciona la opción "Dar de baja". El sistema verifica que no haya vinculaciones activas. Si todo está correcto, desactiva el turno y muestra mensaje de confirmación.
Postcondiciones	El turno queda inactivo.

Tabla 45: Exportar información de turnos

Elemento	Descripción
Código	CU-31
Nombre	Exportar información de turnos
Actor principal	Administrador
Descripción	Permite exportar los datos de turnos registrados a formatos como PDF o Excel.
Precondiciones	Deben existir turnos registrados.
Flujo principal	El administrador accede al listado de turnos. Selecciona la opción "Exportar". El sistema genera el archivo y permite su descarga.
Postcondiciones	El archivo queda disponible para su descarga.
Reglas de negocio	Solo se exportan turnos con base en los filtros aplicados.

Tabla 46: Registrar matrícula

Elemento	Descripción
Código	CU-32
Nombre	Registrar matrícula
Actor principal	Administrador
Descripción	Permite registrar una nueva matrícula asociando estudiante, curso, ciclo académico y turno.
Precondiciones	Deben existir cursos, estudiantes, ciclos académicos y turnos previamente registrados.
Flujo principal	El administrador accede al módulo de matrículas. Selecciona la opción "Registrar nueva matrícula". Selecciona al estudiante, el curso, el turno y el ciclo académico. El sistema valida que no exista una matrícula duplicada con esa combinación. El sistema guarda la nueva matrícula y muestra un mensaje de éxito.
Postcondiciones	Se crea un nuevo registro de matrícula.
Reglas de negocio	No se permite duplicar la combinación de estudiante, curso, ciclo y turno.

Tabla 47: Validar unicidad de matrícula

Elemento	Descripción
Código	CU-33
Nombre	Validar unicidad de matrícula
Actor principal	Sistema
Descripción	Verifica que no exista otra matrícula con la misma combinación de estudiante, curso, turno y ciclo académico.
Precondiciones	Se está intentando registrar o actualizar una matrícula.
Flujo principal	Al registrar o editar una matrícula, el sistema valida la combinación. Si ya existe, muestra mensaje de error y bloquea el registro. Si no existe, permite continuar.
Postcondiciones	Solo se permiten combinaciones únicas.

Tabla 48: Listar matrículas

Elemento	Descripción
Código	CU-34
Nombre	Listar matrículas
Actor principal	Administrador
Descripción	Permite visualizar un listado de todas las matrículas registradas.
Precondiciones	Deben existir matrículas registradas.
Flujo principal	El administrador accede al módulo de matrículas. Visualiza el listado con opciones para buscar, filtrar y ordenar. El sistema muestra la información según los criterios seleccionados.
Postcondiciones	El administrador puede revisar las matrículas existentes.
Reglas de negocio	Se pueden aplicar filtros dinámicos.

Tabla 49: Editar matrícula

Elemento	Descripción
Código	CU-35
Nombre	Editar matrícula
Actor principal	Administrador
Descripción	Permite modificar los datos de una matrícula existente.
Precondiciones	La matrícula debe estar activa y el turno no debe estar cerrado o finalizado.
Flujo principal	El administrador selecciona la matrícula a editar. Realiza los cambios necesarios. El sistema valida y guarda los cambios.
Postcondiciones	La matrícula es actualizada correctamente.
Reglas de negocio	Solo se pueden editar matrículas activas.

Tabla 50: Dar de baja matrícula

Elemento	Descripción
Código	CU-36
Nombre	Dar de baja matrícula
Actor principal	Administrador
Descripción	Permite inactivar o anular una matrícula existente.
Precondiciones	La matrícula debe estar registrada.
Flujo principal	El administrador selecciona una matrícula activa. Da clic en la opción "Dar de baja". El sistema solicita confirmación. Al confirmar, el sistema marca la matrícula como inactiva.
Postcondiciones	La matrícula queda anulada/inactiva.
Reglas de negocio	Las matrículas dadas de baja no deben ser consideradas en reportes activos.

Tabla 51: Exportar información de matrículas

Elemento	Descripción
Código	CU-37
Nombre	Exportar información de matrículas
Actor principal	Administrador
Descripción	Permite exportar la información de las matrículas a formatos compatibles como PDF o Excel.
Precondiciones	Deben existir matrículas registradas.
Flujo principal	El administrador accede al listado de matrículas. Selecciona la opción "Exportar" y permite su descarga.
Postcondiciones	El archivo con la información es descargable.
Reglas de negocio	Solo se exportan las matrículas filtradas en el listado.

Tabla 52: Registrar persona

Elemento	Descripción
Código	CU-38
Nombre	Registrar persona
Actor principal	Administrador
Descripción	Permite registrar a una persona natural dentro del sistema.
Precondiciones	El administrador debe tener acceso al sistema y los campos obligatorios deben estar completos.
Flujo principal	Selecciona la opción "Registrar persona". Ingresa los datos personales. El sistema valida que el documento de identidad sea único. El sistema guarda la información y confirma el registro.
Postcondiciones	La persona queda registrada en el sistema.

Tabla 53: Tipificar persona

Elemento	Descripción
Código	CU-39
Nombre	Tipificar persona
Actor principal	Administrador
Descripción	Permite asignar un tipo a la persona registrada (beneficiario, estudiante o trabajador).
Precondiciones	La persona debe haber sido registrada previamente.
Flujo principal	Selecciona una persona registrada. Asigna un tipo. El sistema valida y guarda el cambio.
Postcondiciones	La persona tiene un tipo asignado.

Tabla 54: Validar unicidad del documento de identidad

Elemento	Descripción
Código	CU-40
Nombre	Validar unicidad del documento de identidad
Actor principal	Sistema
Descripción	Valida que el documento de identidad de la persona no exista previamente en el sistema.
Precondiciones	El administrador está registrando una nueva persona.
Flujo principal	Ingresa el documento de identidad. El sistema verifica si ya está registrado. Si es único, permite continuar. Si ya existe, muestra error.
Postcondiciones	El sistema evita duplicados de documentos de identidad.

Tabla 55: Visualizar lista de personas

Elemento	Descripción
Código	CU-41
Nombre	Visualizar lista de personas
Actor principal	Administrador
Descripción	Permite al administrador visualizar todas las personas registradas en el sistema.
Precondiciones	Debe estar autenticado en el sistema.

Flujo principal	Selecciona "Ver lista de personas". El sistema muestra las personas registradas. Puede buscar, filtrar y ordenar.
Postcondiciones	El administrador visualiza la información según los filtros.

Tabla 56: Editar datos de una persona

Elemento	Descripción
Código	CU-42
Nombre	Editar datos de una persona
Actor principal	Administrador
Descripción	Permite editar los datos de una persona registrada.
Precondiciones	La persona debe estar registrada y activa.
Flujo principal	Selecciona una persona. Realiza los cambios. El sistema valida y guarda la actualización.
Postcondiciones	Los datos de la persona son actualizados.

Tabla 57: Dar de baja a una persona

Elemento	Descripción
Código	CU-43
Nombre	Dar de baja a una persona
Actor principal	Administrador
Descripción	Permite dar de baja a una persona que ya no esté relacionada con el proyecto.
Precondiciones	La persona debe estar registrada.
Flujo principal	Selecciona una persona. Da clic en "Dar de baja". El sistema verifica que no tenga registros activos. El sistema desactiva o elimina la persona.
Postcondiciones	La persona es desactivada y ya no aparece en las listas activas.

Tabla 58: Exportar información de personas

Elemento	Descripción
Código	CU-44
Nombre	Exportar información de personas
Actor principal	Administrador
Descripción	Permite exportar información de personas registradas en formatos como PDF o Excel.
Precondiciones	Debe haber personas registradas y acceso al sistema.
Flujo principal	Selecciona "Exportar información de personas". El sistema genera el archivo. Descarga el archivo.
Postcondiciones	Se obtiene un archivo con la información exportada.

Tabla 59: Registrar tipo de identificación

Elemento	Descripción
Código	CU-45
Nombre	Registrar tipo de identificación
Actor principal	Administrador
Descripción	Permite registrar un nuevo tipo de identificación para personas naturales.
Precondiciones	El administrador debe estar autenticado.
Flujo principal	Accede al módulo de "Tipos de Identificación". Selecciona "Registrar nuevo tipo". Ingresa el nombre. El sistema valida que no exista. Guarda y confirma el registro.
Postcondiciones	Se registra un nuevo tipo de identificación en el sistema.

Tabla 60: Validar unicidad del tipo de identificación

Elemento	Descripción
Código	CU-46
Nombre	Validar unicidad del tipo de identificación
Actor principal	Sistema
Descripción	Valida que no se registren tipos de identificación duplicados.
Precondiciones	El administrador ha ingresado un nuevo tipo.
Flujo principal	El sistema verifica si ya existe uno con el mismo nombre. Si existe, muestra error. Si no existe, permite continuar.
Postcondiciones	Se evita duplicar tipos de identificación.

Tabla 61: Visualizar listado de tipos de identificación

Elemento	Descripción
Código	CU-47
Nombre	Visualizar listado de tipos de identificación
Actor principal	Administrador
Descripción	Permite consultar los tipos de identificación registrados.
Precondiciones	Deben existir tipos registrados.
Flujo principal	Accede al módulo de tipos. El sistema muestra un listado con filtros. Aplica filtros si es necesario.
Postcondiciones	Visualiza los tipos de identificación registrados.

Tabla 62: Editar tipo de identificación

Elemento	Descripción
Código	CU-48
Nombre	Editar tipo de identificación
Actor principal	Administrador
Descripción	Permite modificar los datos de un tipo previamente registrado.
Precondiciones	El tipo no debe estar asociado a procesos cerrados.

Flujo principal	Selecciona el tipo. Realiza los cambios. El sistema valida y guarda.
Postcondiciones	El tipo de identificación es actualizado.

Tabla 63: Dar de baja tipo de identificación

Elemento	Descripción
Código	CU-49
Nombre	Dar de baja tipo de identificación
Actor principal	Administrador
Descripción	Permite desactivar tipos que ya no se utilizan.
Precondiciones	El tipo no debe tener relaciones activas.
Flujo principal	Accede al listado. Selecciona el tipo. Da click en desactivar. El sistema marca como inactivo.
Postcondiciones	El tipo ya no aparece como disponible en procesos activos.

Tabla 64: Exportar información de tipos de identificación

Elemento	Descripción
Código	CU-50
Nombre	Exportar información de tipos de identificación
Actor principal	Administrador
Descripción	Permite exportar los tipos registrados en PDF o Excel.
Precondiciones	Deben existir tipos registrados.
Flujo principal	Accede al listado. Selecciona "Exportar". El sistema genera y habilita la descarga.
Postcondiciones	Se obtiene un archivo con los tipos de identificación registrados.

Tabla 65: Registrar asistencia de estudiantes

Elemento	Descripción
Código	CU-51
Nombre	Registrar asistencia de estudiantes
Actor principal	Administrador / Docente
Descripción	Permite registrar la asistencia diaria de los estudiantes a los cursos.
Precondiciones	Los estudiantes deben estar previamente inscritos en cursos activos.
Flujo principal	Accede al módulo de asistencia. Filtra por curso, ciclo académico y fecha. Lista estudiantes. Marca asistencia. Confirma y guarda.
Postcondiciones	Se registra la asistencia de los estudiantes para el día indicado.

Tabla 66: Validar asistencia por día

Elemento	Descripción
Código	CU-52
Nombre	Validar asistencia por día
Actor principal	Sistema
Descripción	Verifica que no se duplique el registro de asistencia para un mismo estudiante, curso y día.
Precondiciones	El estudiante debe estar inscrito en un curso activo.
Flujo principal	El sistema valida si ya existe un registro. Si existe, muestra advertencia. Si no, permite el registro.
Postcondiciones	Se evita el registro duplicado de asistencia.
Postcondiciones	Se facilita la marcación de asistencia con filtros personalizados.

Tabla 67: Visualizar lista de estudiantes para asistencia

Elemento	Descripción
Código	CU-53
Nombre	Visualizar lista de estudiantes para asistencia
Actor principal	Administrador / Docente
Descripción	Permite listar a los estudiantes filtrados por criterios específicos para facilitar la marcación.
Precondiciones	Deben existir estudiantes inscritos en cursos activos.
Flujo principal	Accede al módulo de asistencia. Selecciona filtros de búsqueda. Muestra lista correspondiente.

Tabla 68: Editar asistencia

Elemento	Descripción
Código	CU-54
Nombre	Editar asistencia
Actor principal	Administrador / Docente
Descripción	Permite modificar registros de asistencia guardados en caso de errores o cambios.
Precondiciones	Debe existir un registro de asistencia previamente guardado.
Flujo principal	Accede al módulo. Busca por fecha, estudiante o curso. Selecciona y edita. Guarda los cambios.
Postcondiciones	El registro de asistencia se actualiza correctamente.

Tabla 69: Exportar información de asistencia

Elemento	Descripción
Código	CU-55
Nombre	Exportar información de asistencia
Actor principal	Administrador

Descripción	Permite exportar los registros de asistencia en PDF o Excel.
Precondiciones	Deben existir registros de asistencia en el sistema.
Flujo principal	Accede al módulo. Aplica filtros. Selecciona "Exportar". El sistema genera el archivo.
Postcondiciones	El archivo queda disponible para su descarga y uso externo.

Tabla 70: Registrar notas escolares

Elemento	Descripción
Código	CU-56
Nombre	Registrar notas escolares
Actor principal	Administrador / Docente
Descripción	Permite ingresar las calificaciones de los estudiantes en un curso, divididas en cuatro evaluaciones del 25% cada una.
Precondiciones	Deben existir cursos, estudiantes y ciclos académicos activos registrados.
Flujo principal	Accede al módulo de notas. Selecciona curso, ciclo y estudiante. Ingresa las 4 notas. El sistema valida y guarda. Muestra mensaje de éxito.
Postcondiciones	Las notas quedan registradas correctamente en el sistema.

Tabla 71: Validar duplicidad de notas

Elemento	Descripción
Código	CU-57
Nombre	Validar duplicidad de notas
Actor principal	Sistema
Descripción	Garantiza que no se registre más de un conjunto de notas para el mismo estudiante, curso y ciclo.
Precondiciones	El estudiante debe estar inscrito en el curso y ciclo.
Flujo principal	El sistema verifica si ya hay notas. Si existen, bloquea el registro y notifica. Si no, permite registrar.
Postcondiciones	Se evita la duplicidad de registros de notas.

Tabla 72: Visualizar notas escolares

Elemento	Descripción
Código	CU-58
Nombre	Visualizar notas escolares
Actor principal	Administrador / Docente
Descripción	Permite consultar notas registradas por estudiante, curso y ciclo académico.
Precondiciones	Deben existir notas registradas.

Flujo principal	Accede al listado de notas. Aplica filtros (curso, ciclo, estudiante, fecha). El sistema muestra la lista filtrada.
Postcondiciones	Se visualiza la información académica de manera clara y ordenada.

Tabla 73: Editar notas escolares

Elemento	Descripción
Código	CU-59
Nombre	Editar notas escolares
Actor principal	Administrador / Docente
Descripción	Permite modificar los registros de notas en caso de corrección o ajuste académico.
Precondiciones	El estudiante debe tener notas previamente registradas.
Flujo principal	Busca el registro a modificar. Accede al formulario de edición. Realiza cambios. El sistema valida y guarda.
Postcondiciones	El sistema actualiza los valores de notas correctamente.

Tabla 74: Exportar notas escolares

Elemento	Descripción
Código	CU-60
Nombre	Exportar notas escolares
Actor principal	Administrador / Docente
Descripción	Permite generar un reporte exportable con los registros de notas académicas.
Precondiciones	Deben existir notas registradas en el sistema.
Flujo principal	Accede a la sección de notas. Selecciona "Exportar". El sistema genera y descarga el archivo.
Postcondiciones	El archivo queda disponible para uso externo.

Tabla 75: Registrar consulta médica o psicológica

Elemento	Descripción
Código	CU-61
Nombre	Registrar consulta médica o psicológica
Actor principal	Administrador
Descripción	Permite registrar una nueva consulta médica o psicológica para un estudiante, beneficiario o trabajador.
Precondiciones	Debe existir al menos una persona registrada en el sistema.
Flujo principal	Accede al módulo de consultas. Selecciona "Registrar consulta". Llena los datos. El sistema valida. Guarda y confirma el registro.
Postcondición	La consulta queda registrada en el sistema.

Tabla 76: Validar unicidad de nombre de consulta

Elemento	Descripción
Código	CU-62
Nombre	Validar unicidad de nombre de consulta
Actor principal	Sistema
Descripción	Valida automáticamente que el nombre de la consulta no exista previamente en el sistema.
Precondiciones	El administrador debe estar registrando una nueva consulta.
Flujo principal	El sistema compara el nombre con los existentes. Si ya existe, muestra error.

Tabla 77: Visualizar listado de consultas

Elemento	Descripción
Código	CU-63
Nombre	Visualizar listado de consultas
Actor principal	Administrador
Descripción	Permite visualizar todas las consultas registradas, con funciones de búsqueda y filtrado.
Precondiciones	Deben existir consultas registradas.
Flujo principal	Accede al módulo de consultas. Se muestra listado con filtros. Aplica filtros según necesidad.
Postcondiciones	El listado de consultas se presenta con claridad.

Tabla 78: Editar consulta

Elemento	Descripción
Código	CU-64
Nombre	Editar consulta
Actor principal	Administrador
Descripción	Permite modificar los datos de una consulta registrada.
Precondiciones	La consulta debe existir y estar activa.
Flujo principal	Localiza la consulta en el listado. Selecciona "Editar". Ajusta los datos. El sistema actualiza y confirma.
Postcondiciones	Los cambios quedan aplicados correctamente.

Tabla 79: Exportar información de consultas

Elemento	Descripción
Código	CU-65
Nombre	Exportar información de consultas
Actor principal	Administrador
Descripción	Permite exportar los datos de consultas en formatos como PDF o Excel.
Precondiciones	Deben existir consultas registradas.

Flujo principal	Accede al listado de consultas. Selecciona "Exportar". El sistema genera y permite descargar el archivo.
Postcondiciones	El archivo queda disponible para uso externo.

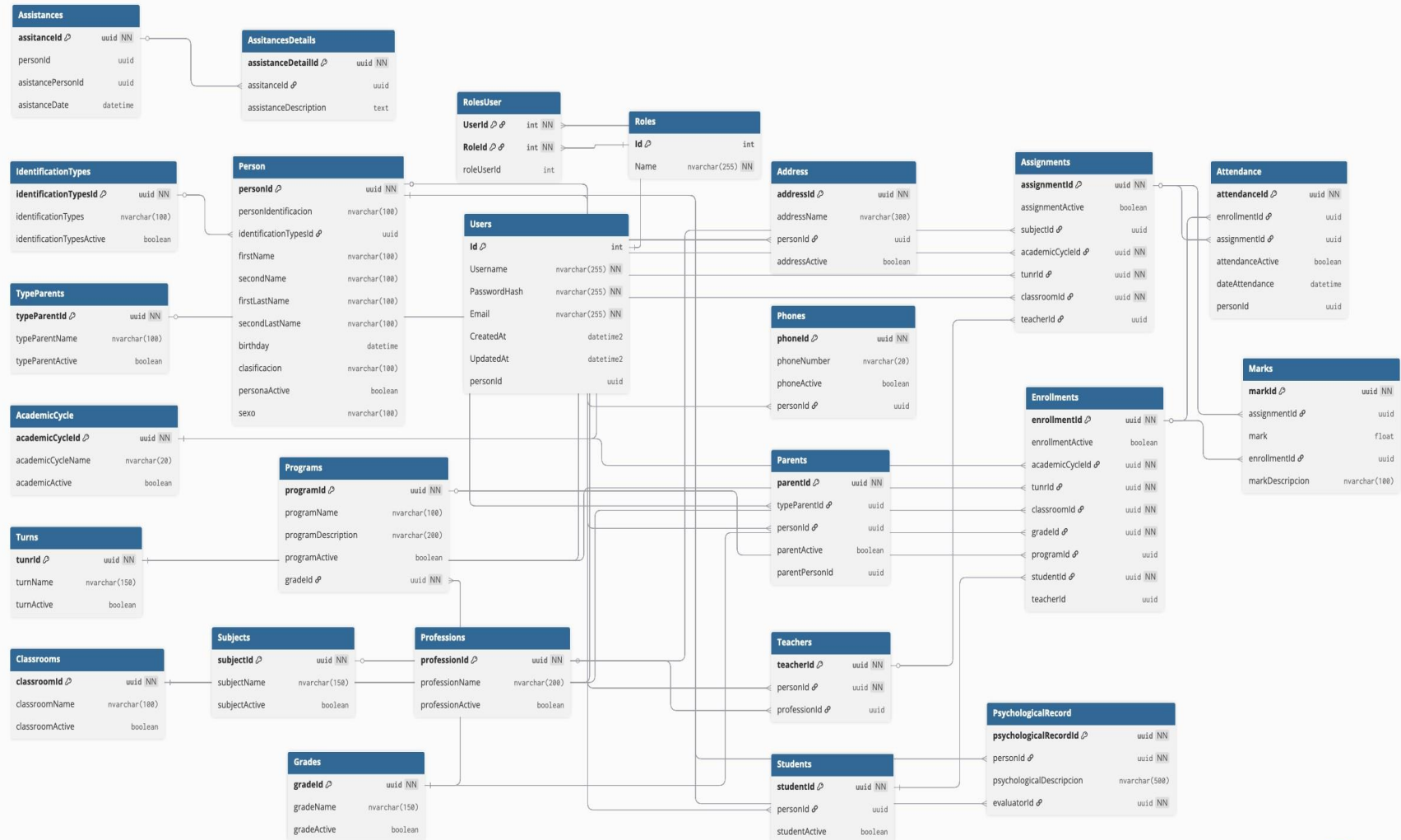
7.7. DISEÑO DEL MODELO ENTIDAD RELACIÓN

El objetivo de este proceso es definir la arquitectura del sistema y el entorno tecnológico que le dará soporte. Esta tarea se abordó durante el Sprint 2, correspondiente a las semanas 3 y 4 del proceso Scrum, donde se avanzó significativamente en el diseño técnico del sistema. En esta etapa se realizó la configuración del entorno de desarrollo, se seleccionaron las tecnologías base y se definieron los componentes principales de la arquitectura, sentando las bases para el desarrollo modular y escalable del sistema PROYSMAPA.

Como parte fundamental de este sprint, se llevó a cabo el modelado de datos mediante el Diagrama Entidad-Relación (ER), el cual permitió representar de manera estructurada las entidades clave del sistema, sus atributos y las relaciones existentes entre ellas. Este modelo fue refinado a partir del análisis de requerimientos y del trabajo colaborativo con los stakeholders, y forma parte del conjunto de diagramas UML utilizados para visualizar la estructura del sistema.

El Diagrama ER no solo facilita la comprensión del dominio del problema, sino que también constituye la base para el diseño lógico y físico de la base de datos, asegurando la coherencia e integridad de la información durante todo el ciclo de vida del sistema. Además, su implementación efectiva en el Sprint 2 permitió generar especificaciones claras para la construcción y codificación de los módulos funcionales en los siguientes sprints.

Figura 2: Modelo Entidad Relación



7.8. DISEÑO DE DIAGRAMAS DE ACTIVIDADES

El diagrama de actividades es una herramienta de modelado fundamental en la ingeniería de software, utilizada para representar de manera gráfica el flujo de actividades y decisiones dentro de un proceso o sistema. Este tipo de diagrama resulta especialmente útil para visualizar el comportamiento dinámico de un sistema, identificar los pasos de un flujo de trabajo y comprender la lógica detrás de la toma de decisiones, facilitando así el análisis y diseño de procesos funcionales.

Nota. Las siguientes figuras son elaboración propia.

Figura 3: Diagrama de Actividades - Login

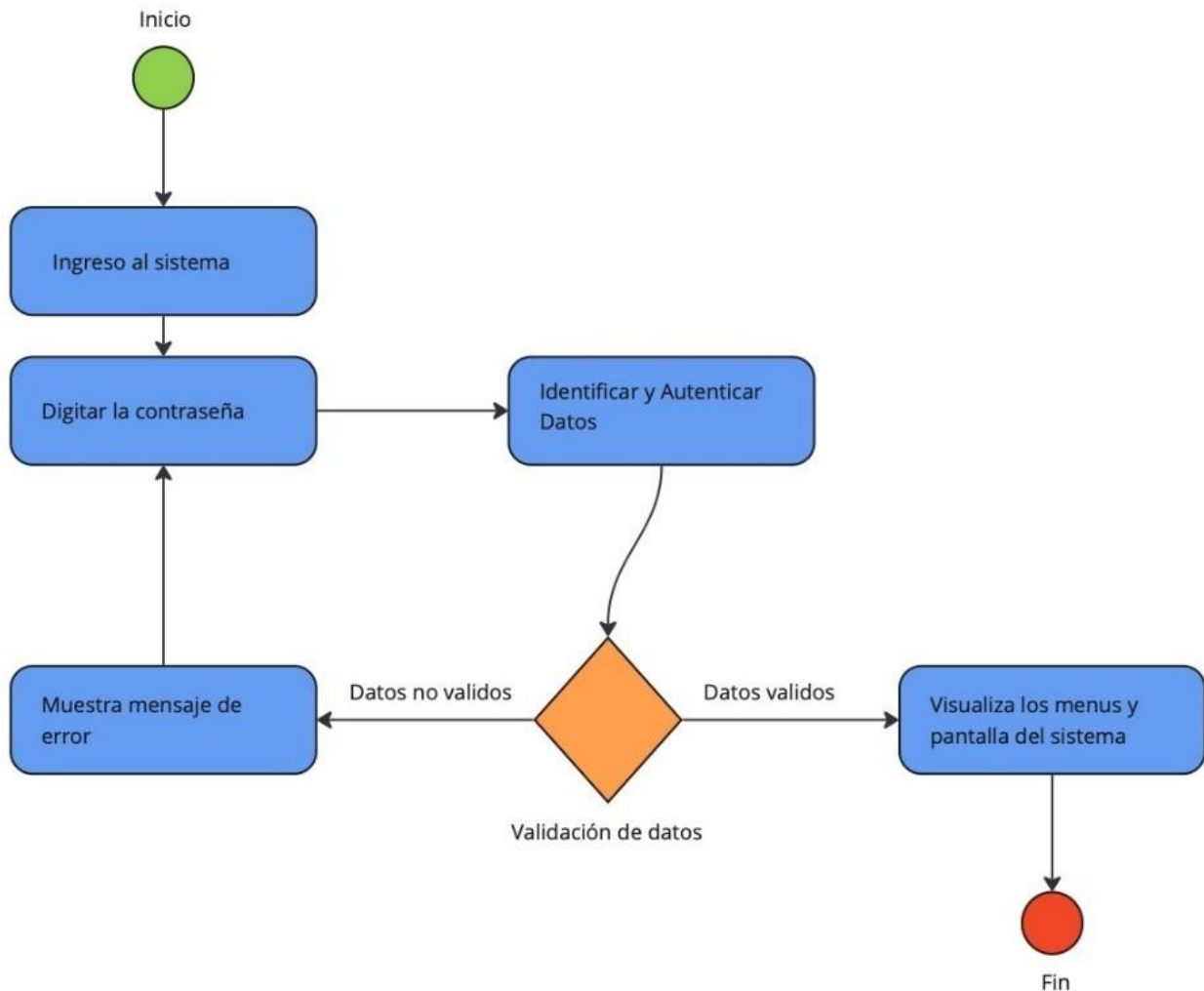


Figura 4: Diagrama de Actividades – Profesiones

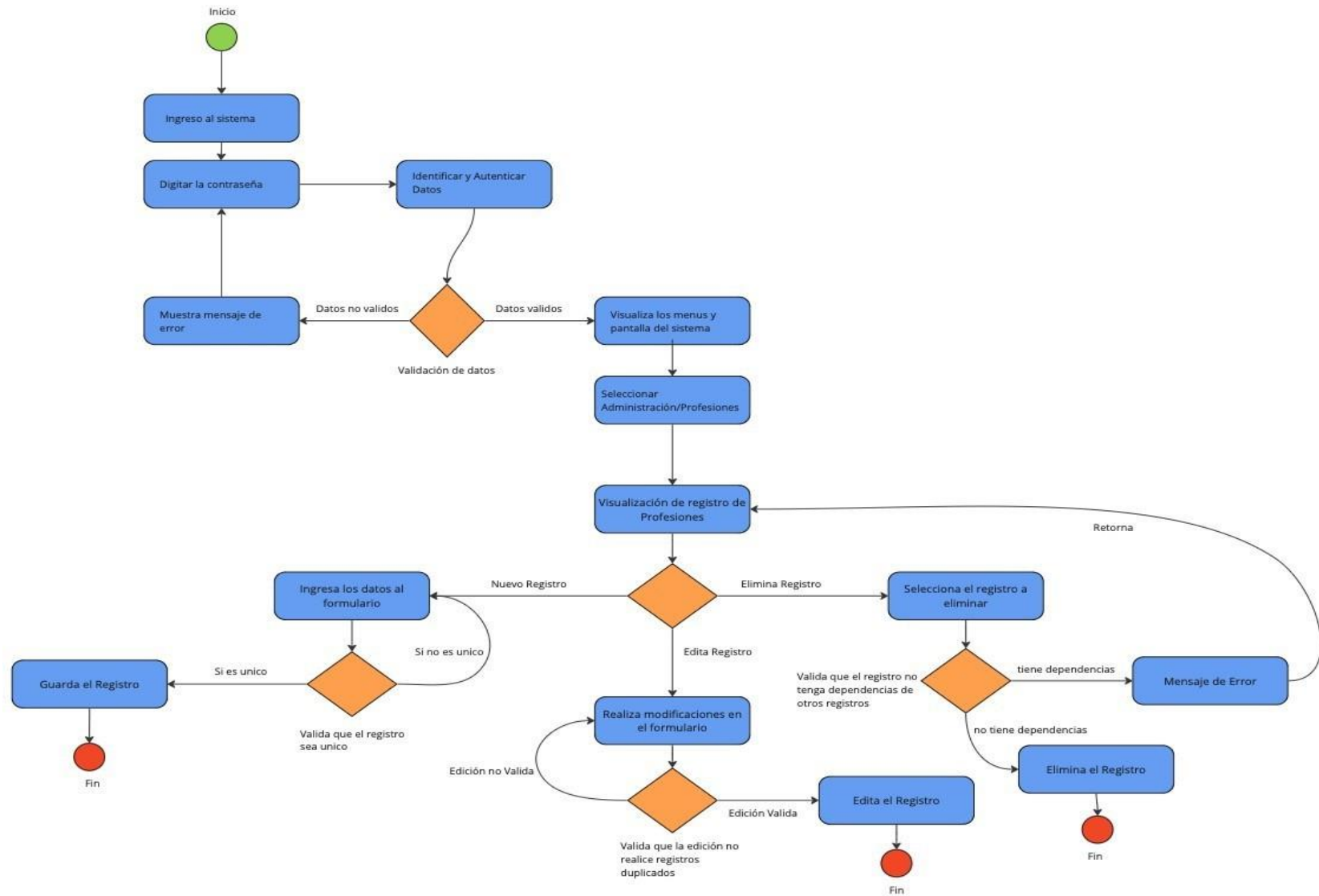


Figura 5: Diagrama de Actividades – Aulas

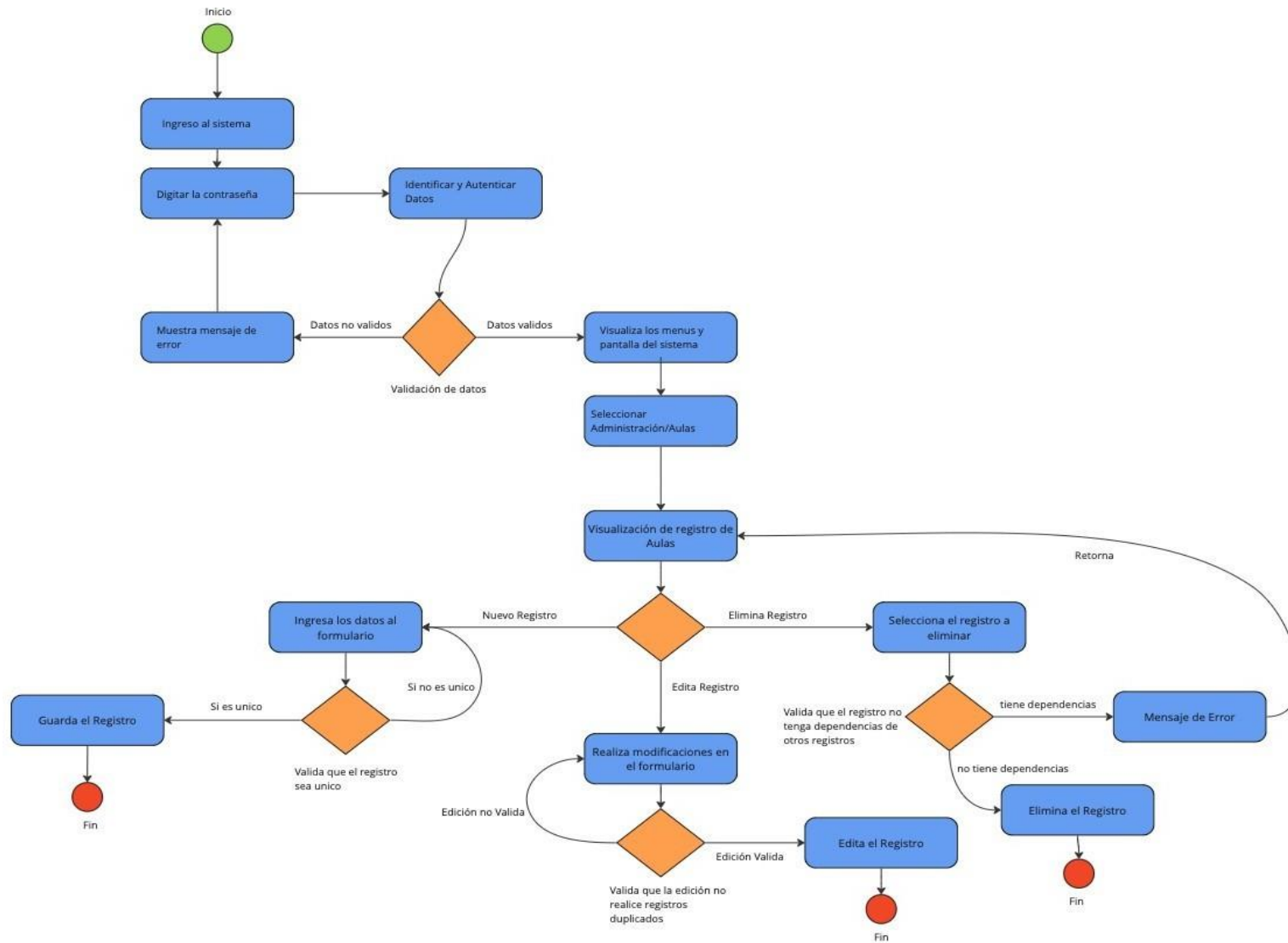


Figura 6: Diagrama de Actividades – Ciclo Académico

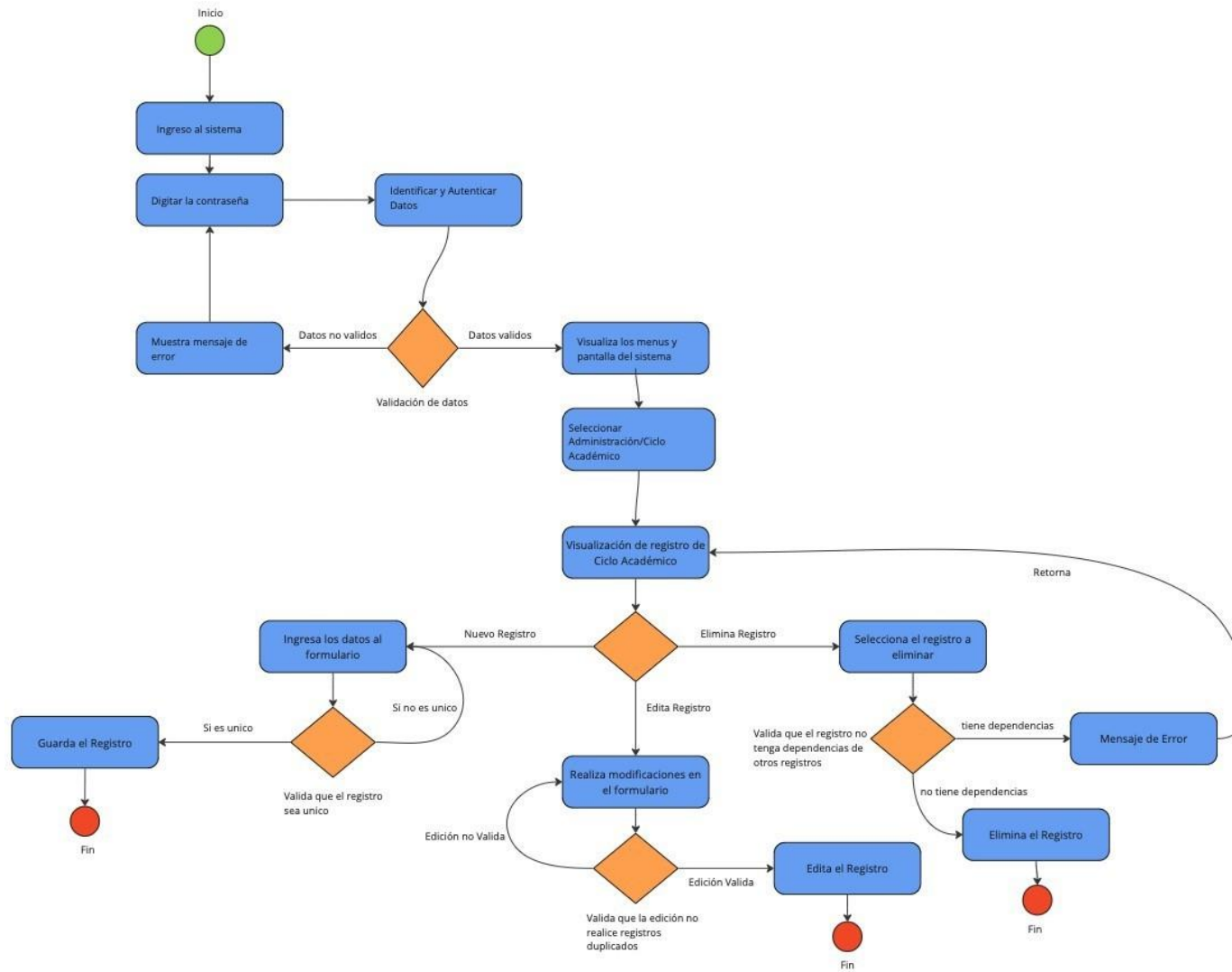


Figura 7: Diagrama de Actividades – Cursos

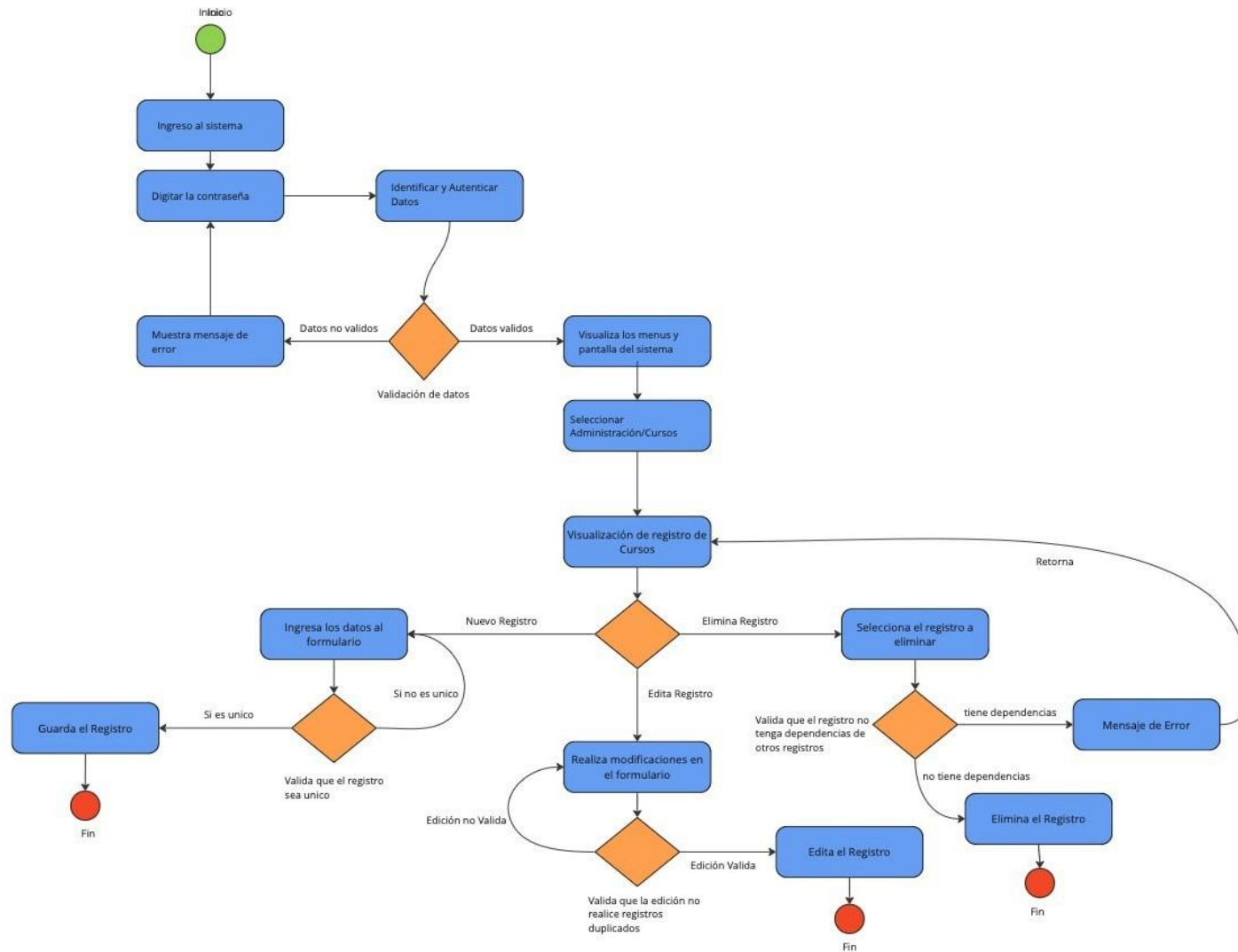


Figura 8: Diagrama de Actividades – Turnos

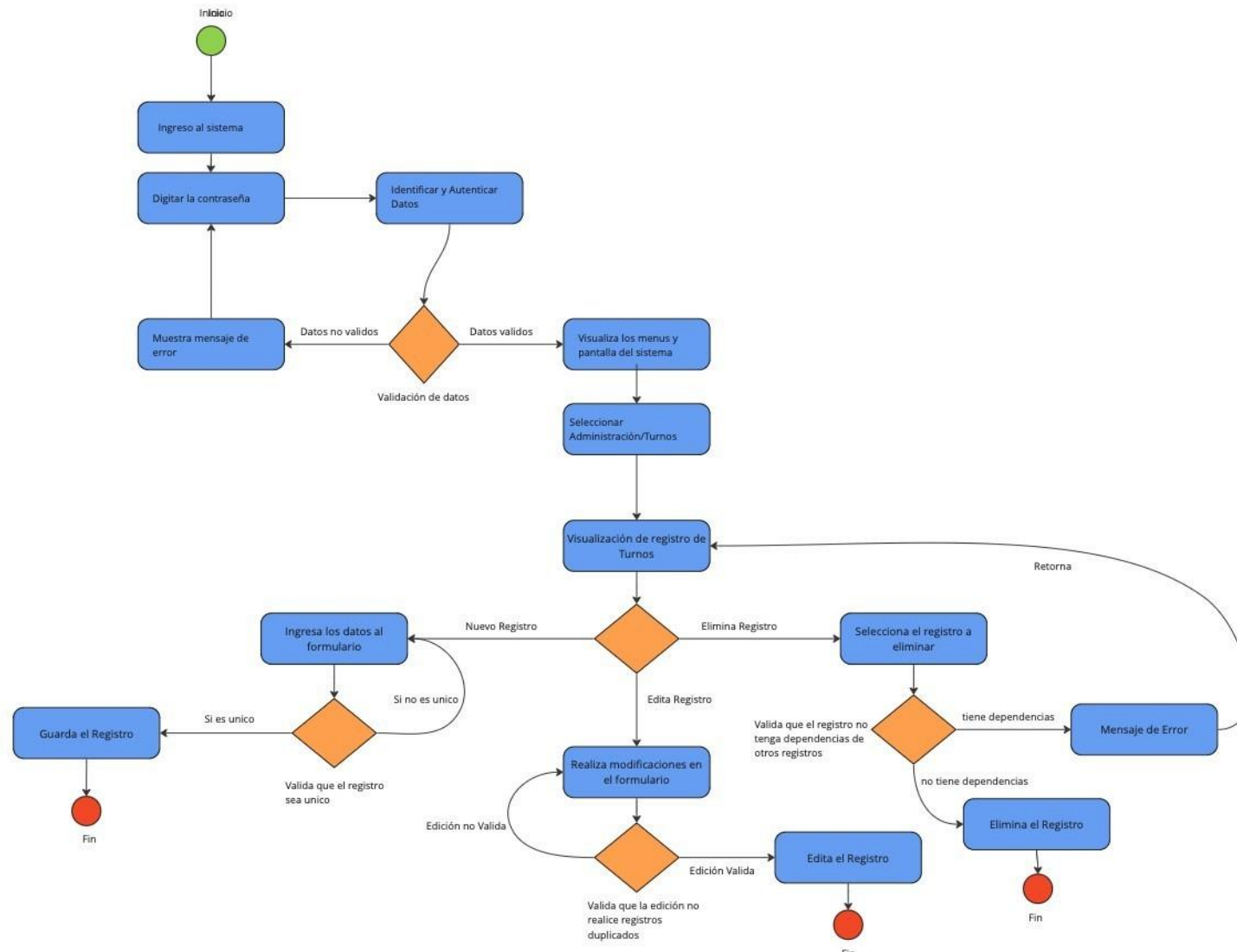


Figura 9: Diagrama de Actividades – Matriculas

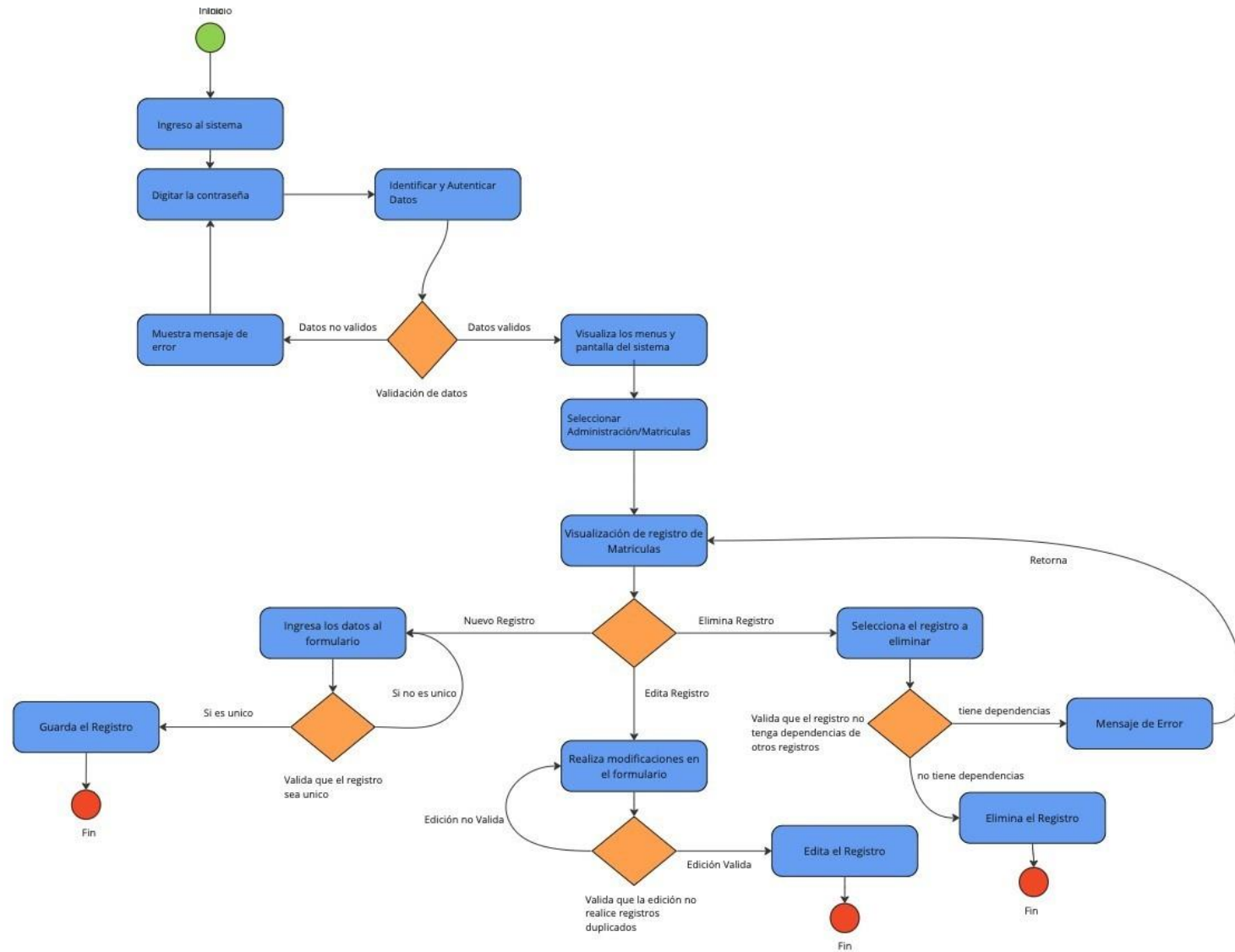


Figura 10: Diagrama de Actividades – Registro Persona

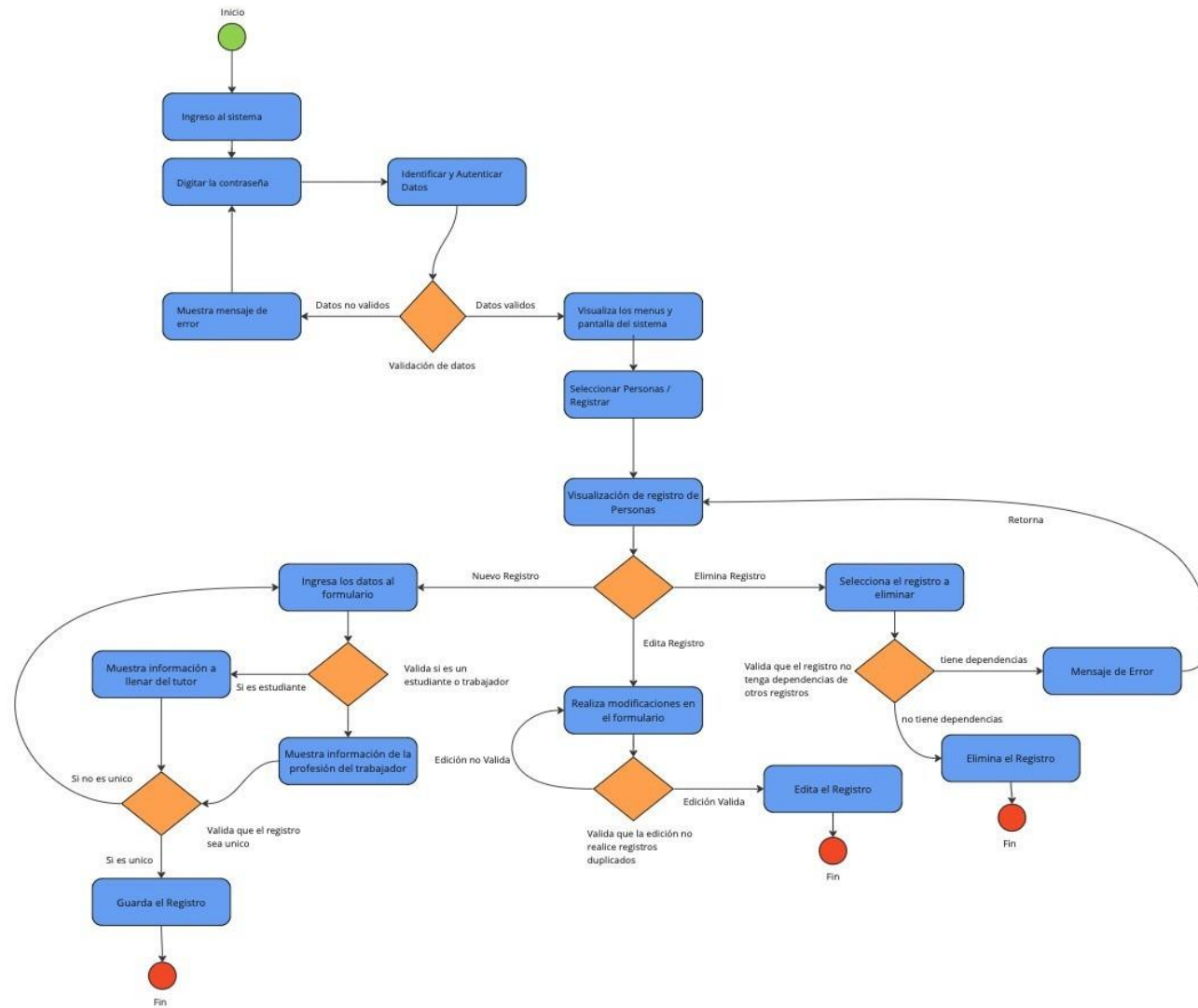


Figura 11: Diagrama de Actividades – Tipo Identificación

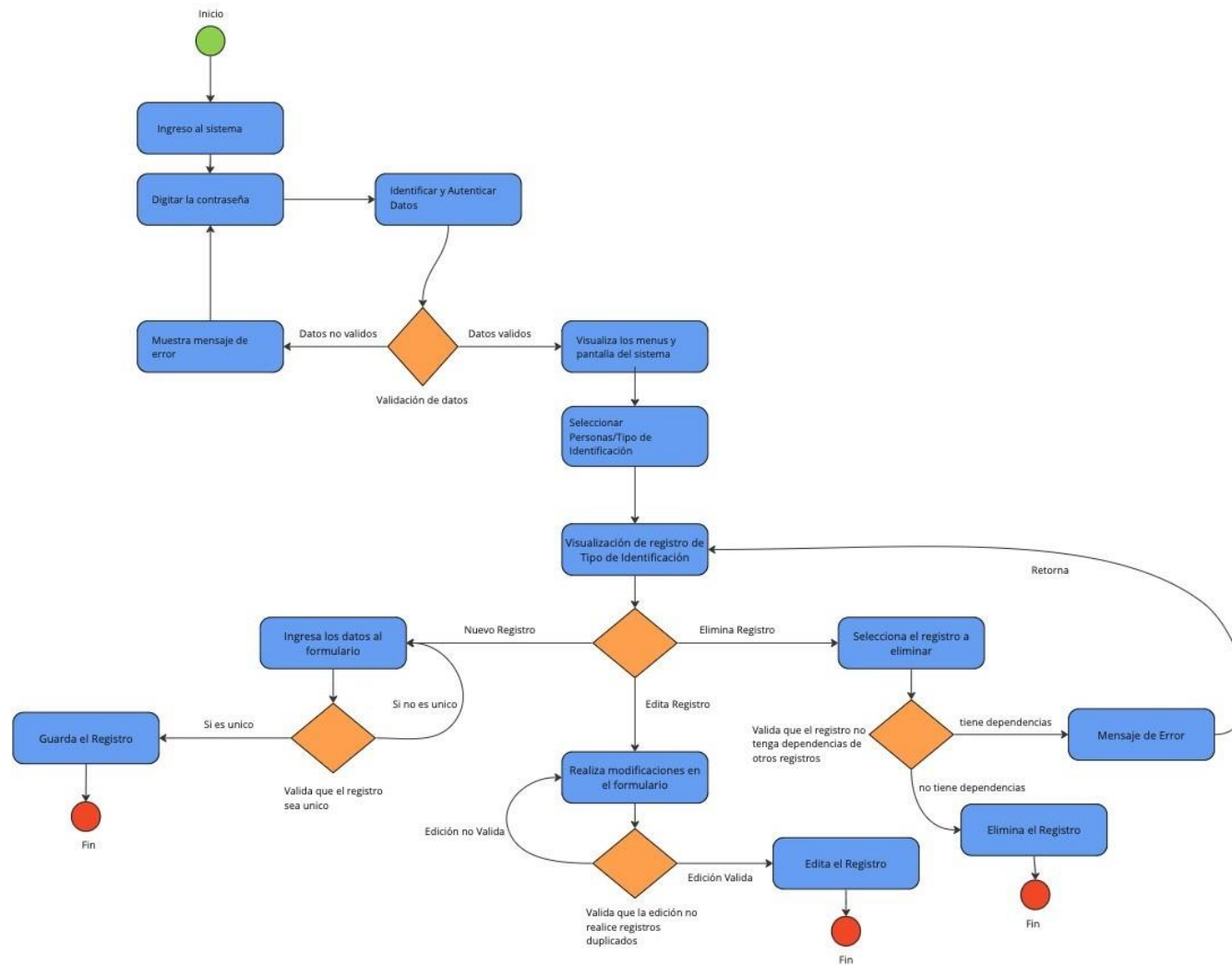


Figura 12: Diagrama de Actividades – Asistencia Escolar

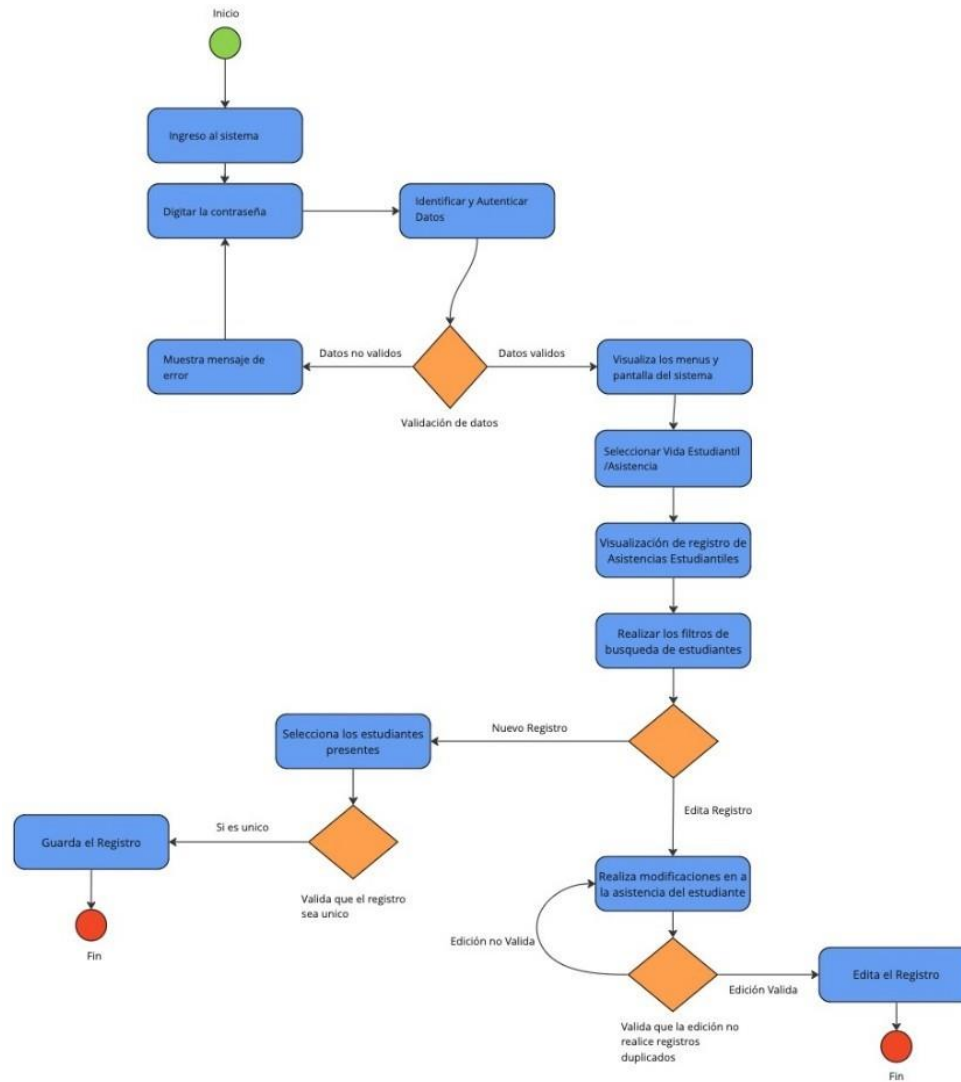


Figura 13: Diagrama de Actividades – Notas

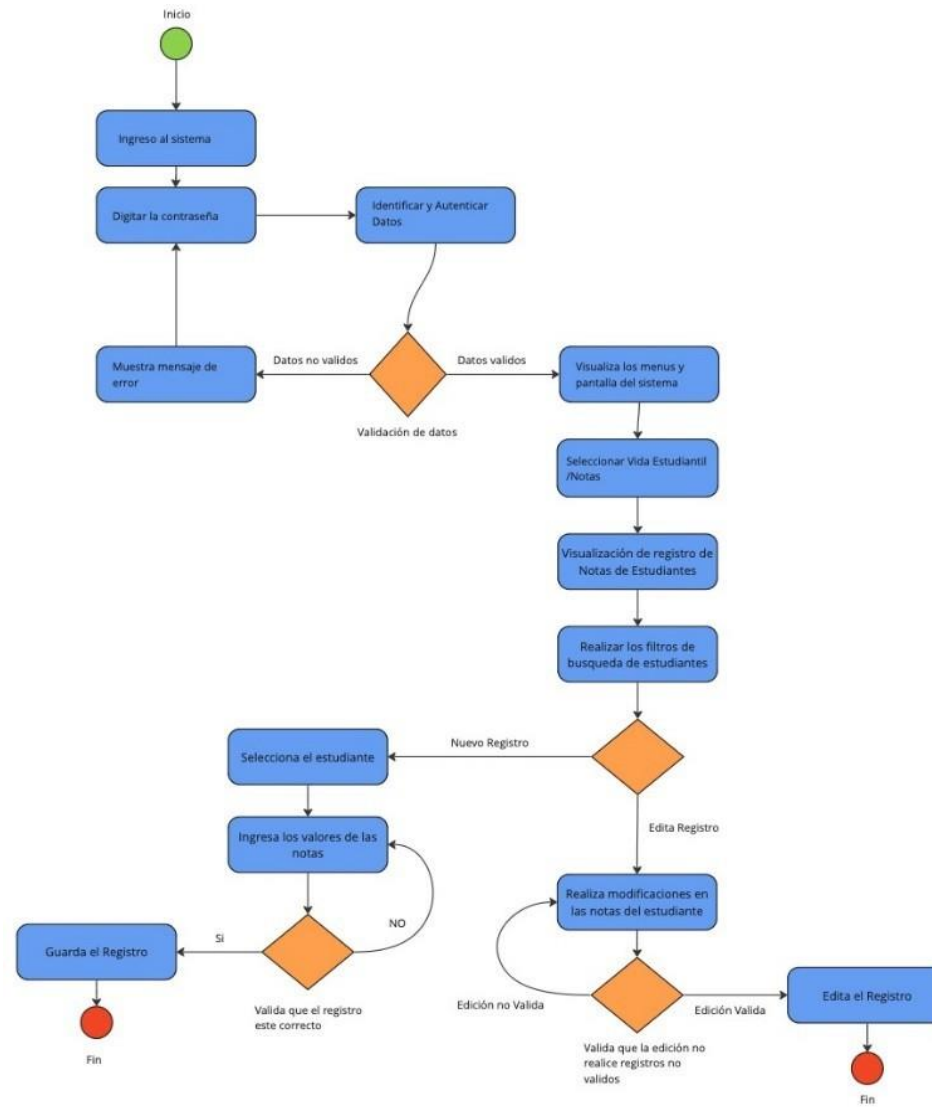
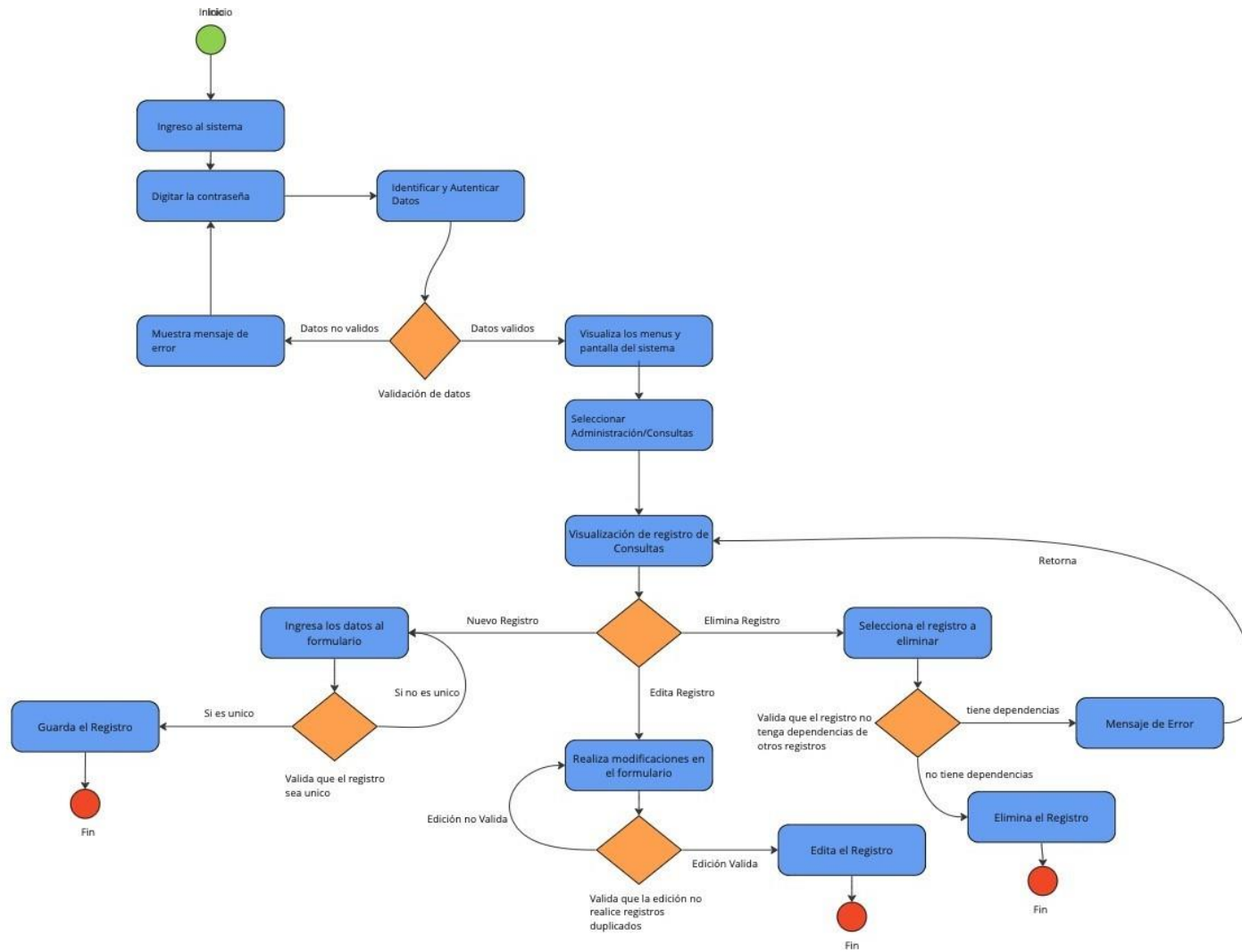


Figura 14: Diagrama de Actividades – Consultas Médicas o Psicológicas



7.9. DISEÑO DE LA INTERFAZ DE USUARIO DEL SISTEMA

El diseño de la interfaz del sistema PROYSMAPA fue desarrollado con un enfoque centrado en la experiencia del usuario (UX), priorizando la usabilidad, accesibilidad y estética visual. A lo largo de varios sprints se abordaron diferentes etapas del diseño e implementación de la interfaz, iniciando en el Sprint 3 (semanas 5-6) con la creación de interfaces básicas para los módulos iniciales, como gestión de usuarios y actividades. En el Sprint 5 (semanas 9-10) se realizaron ajustes y revisiones al diseño inicial, incorporando mejoras visuales y funcionales basadas en los avances obtenidos durante el desarrollo.

Posteriormente, durante el Sprint 7 (semanas 13-14), se finalizó la implementación de las funcionalidades relacionadas con los usuarios y se llevaron a cabo pruebas de usabilidad, lo que permitió refinar los elementos visuales y la organización de los componentes de la interfaz para garantizar una navegación intuitiva. Se emplearon plantillas de administración modernas, responsivas y compatibles con dispositivos móviles, lo cual no solo optimiza la presentación visual, sino que también reduce la posibilidad de errores operativos.

Nota. Las siguientes figuras son elaboración propia.

Figura 15: Login del Sistema

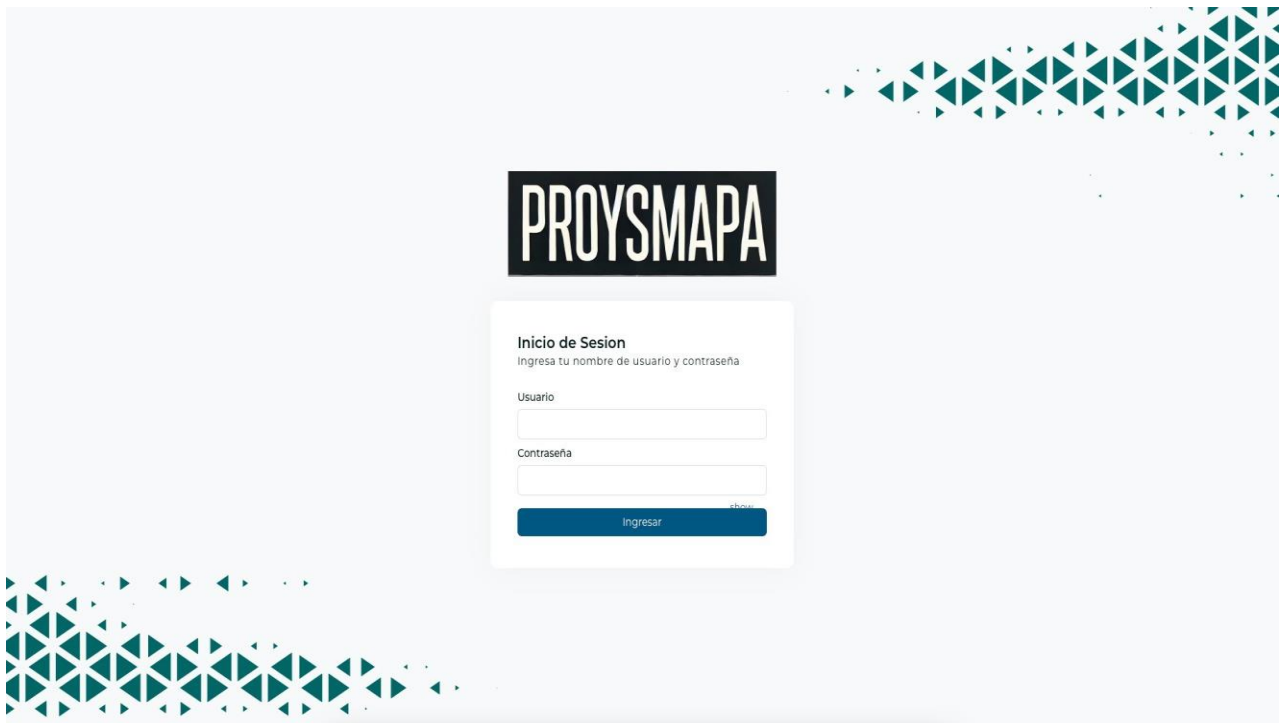
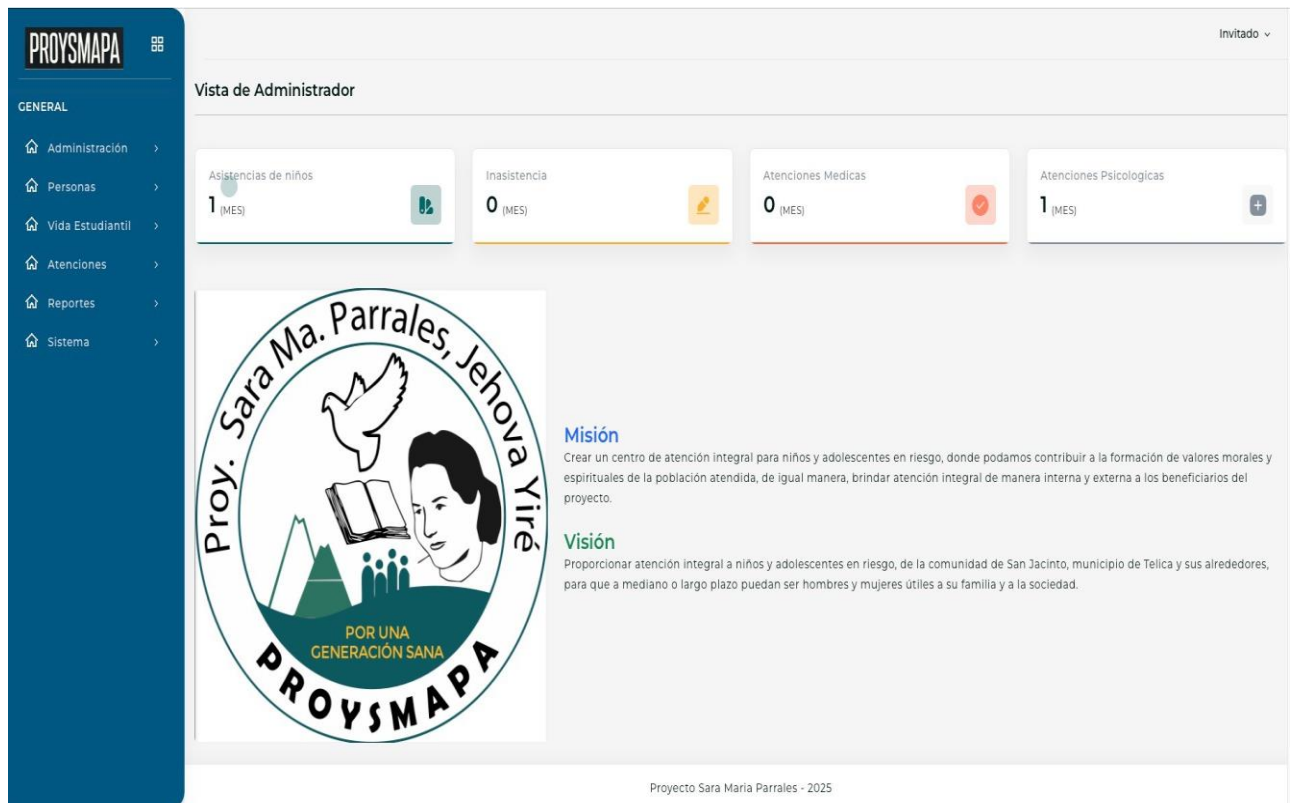


Figura 16: Menú Principal del Sistema



Una vez autenticado, el sistema presenta un menú principal personalizado según el rol del usuario. Desde este menú se puede acceder fácilmente a los diferentes módulos del sistema, así como visualizar la información de la cuenta activa.

Figura 17: Formularios del Sistema

Registro de Persona

Primer Nombre
Segundo Nombre

Primer Apellido
Segundo Apellido

Tipo de Identificación
Numero identificación

Sexo
Fecha de Nacimiento
20/04/2025

Clasificación
Telefono

Direccion:

Guardar Cancelar

Personas
Listado de Personas

Primer Nombre	Segundo Nombre	Primer Apellido	Segundo Apellido	Identificación
Profesor				1321232343242f
Byron				123123123123123g
Byron				566726373h

Mostrando 1 de 3 hasta 3 Registros

Proyecto Sara Maria Parrales - 2025

Los formularios están diseñados para facilitar el ingreso de datos alfanuméricos mediante campos de texto claros y bien organizados. Se incluye validación de datos para garantizar la integridad de la información registrada.

Figura 18: Tablas del Sistema

The screenshot displays the PROYSMAPA web application. On the left is a dark blue sidebar menu with the following items: GENERAL, Administración, Personas, Vida Estudiantil (selected), Exp. Estudiantes, Asistencia, Notas, Atenciones, Reportes, and Sistema. The main content area is titled 'Lista de estudiantes' and 'Listado de Personas'. It features a 'Registrar Asistencia' button in the top right. Below the title are three filters: 'Aula' (dropdown with 'A2'), 'Turno' (dropdown with 'Matutino'), and 'Fecha' (text input with '20/04/2025'). A 'Buscar' button is next to a search input field. Below these is a table with the following columns: Primer Nombre, Segundo Nombre, Primer Apellido, Segundo Apellido, Identificacion, and Asistencia. The table contains one record for a student named Byron Segundo Hijo. At the bottom of the table, it says 'Mostrando 1 de 1 hasta 1 Registros'. The footer of the page reads 'Proyecto Sara María Parrales - 2025'.

Primer Nombre	Segundo Nombre	Primer Apellido	Segundo Apellido	Identificacion	Asistencia
Byron	Segundo	Hijo	Hijo	123123123123123g	☐

Las tablas permiten mostrar de manera ordenada y paginada los registros del sistema e incluyen funciones de búsqueda, filtrado y ordenamiento, lo que mejora la consulta y gestión de la información almacenada.

7.10. CODIFICACIÓN DEL SISTEMA

El desarrollo del módulo administrativo del sistema PROYSMAPA se llevó a cabo utilizando ASP.NET Core 8 como tecnología base, implementando el patrón de diseño Modelo-Vista- Controlador (MVC). Este patrón es ampliamente reconocido en el ámbito del desarrollo de aplicaciones web modernas por su capacidad para separar las distintas responsabilidades del software, lo que favorece la escalabilidad, el mantenimiento y la reutilización del código.

Fundamentos del Patrón MVC

El patrón MVC divide la aplicación en tres componentes fundamentales:

Modelo (Model):

Esta capa representa el núcleo de la lógica de negocio y el acceso a los datos. Se encarga de interactuar con la base de datos, gestionar las reglas de negocio y procesar la información que será presentada al usuario. En el caso del sistema PROYSMAPA, los modelos se implementaron como clases fuertemente tipadas que representan entidades tales como Usuarios, Roles, Proyectos, Beneficiarios, entre otros. ASP.NET Core 8 ofrece integración nativa con Entity Framework Core, lo cual facilitó el uso de técnicas ORM (Object-Relational Mapping) para la manipulación de datos.

Vista (View):

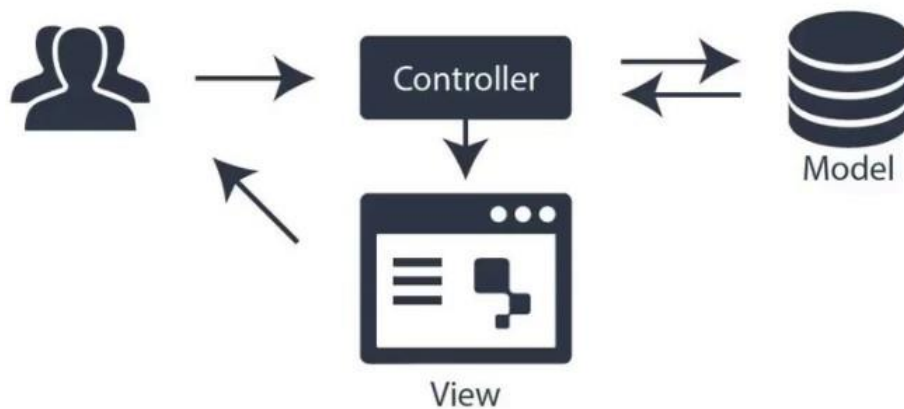
La vista corresponde a la interfaz de usuario. Su función principal es presentar los datos al usuario de forma clara y estructurada. En ASP.NET Core, se utilizó Razor Pages para generar dinámicamente el contenido HTML en función de los datos enviados por el controlador. Se priorizó la separación de la lógica de presentación y la lógica de negocio, asegurando así una mayor mantenibilidad del código.

Controlador (Controller):

El controlador actúa como puente entre la vista y el modelo. Es responsable de recibir las solicitudes HTTP, procesarlas, interactuar con los modelos necesarios y retornar la respuesta adecuada, ya sea una vista o datos en formato JSON. En el módulo administrativo de PROYSMAPA, los controladores fueron organizados en áreas específicas para mejorar la modularidad y seguridad del sistema, como, por ejemplo: AdminController, ProyectoController, y UsuarioController.

Figura 19: Modelo MVC

Model-View-Controller



Medium.com / Joseph Spinelli (2018). MVC Overview. Recuperado de [\[https://71medium.com/@joespinelli_6190/mvc-model-view-controller-ef878e2fd6f5\]](https://71medium.com/@joespinelli_6190/mvc-model-view-controller-ef878e2fd6f5)

- **Ventajas del Enfoque Utilizado**

La elección de ASP.NET Core 8 junto al patrón MVC trajo consigo una serie de beneficios significativos para el desarrollo del sistema PROYSMAPA:

- **Separación de responsabilidades:** Facilita el trabajo en equipo al permitir que diseñadores, desarrolladores de backend y frontend trabajen de forma paralela en la misma aplicación sin interferencias.
- **Reutilización de componentes:** Se diseñaron vistas parciales y servicios reutilizables que optimizan el rendimiento del desarrollo.

- **Escalabilidad:** La arquitectura modular permite incorporar nuevas funcionalidades al módulo administrativo sin alterar significativamente las existentes.
- **Mantenimiento simplificado:** La estructura clara del código y la documentación generada facilitan futuras modificaciones o adaptaciones del sistema.

En resumen, la implementación del módulo administrativo en ASP.NET Core 8 con el patrón MVC resultó ser una decisión técnica acertada, permitiendo construir un sistema sólido, flexible y preparado para futuras ampliaciones. Esta base técnica robusta asegura que el sistema PROYSMAPA pueda mantenerse y escalarse en el tiempo de acuerdo con las necesidades del proyecto.

7.11. PRUEBAS UNITARIAS

Durante la fase de desarrollo del sistema PROYSMAPA, se incorporó una etapa de pruebas unitarias con el objetivo de garantizar la correcta funcionalidad de los distintos módulos administrativos que lo componen. Estas pruebas fueron realizadas utilizando el framework MSTest, una herramienta integrada en el entorno de desarrollo de Visual Studio que permite definir, ejecutar y analizar pruebas de manera estructurada.

La ejecución de pruebas unitarias es una práctica fundamental dentro de metodologías ágiles como Scrum, ya que permite verificar de forma temprana y continua el comportamiento esperado del software, reduciendo la posibilidad de errores en etapas avanzadas del ciclo de vida del sistema.

- **Herramientas Utilizadas**

Lenguaje de programación: C# (.NET 8)

Framework de pruebas: MSTest (Microsoft.VisualStudio.TestTools.UnitTesting)

Entorno de desarrollo: Visual Studio 2022

- **Casos de Prueba**

A continuación, se resumen algunos de los casos de prueba más representativos implementados:

Nota. Las siguientes tablas son elaboración propia.

Tabla 80: Casos de Prueba – Profesiones

Caso de Esperado Prueba	Descripción	Datos de Entrada	Resultado
CP01	Registrar una nueva profesión	"Psicólogo"	Profesión registrada exitosamente
CP02	Registrar profesión duplicada	"Psicólogo" ya existente	Mensaje de error: profesión ya registrada
CP03	Asociar profesión a persona existente	ID profesión: 1, ID persona: 5	Asociación exitosa
CP04	Listar profesiones activas	-	Lista visible de profesiones activas
CP05	Editar profesión existente	ID: 1, nuevo nombre: "Psicoterapeuta"	Cambios guardados correctamente
CP06	Eliminar profesión asociada a persona activa	ID: 2	Mensaje de error: la profesión está asociada
CP07	Exportar lista de profesiones	Excel	Archivo generado correctamente

Tabla 81: Casos de Prueba – Aulas

CP	Descripción	Entrada	Resultado Esperado
CP08	Registrar aula nueva	"Aula 101"	Aula registrada
CP09	Registrar aula duplicada	"Aula 101"	Mensaje de error
CP10	Asociar aula a curso	ID aula: 1, ID curso: 3	Asociación exitosa
CP11	Listar aulas activas	-	Lista mostrada
CP12	Editar aula	ID: 1, nuevo nombre: "Laboratorio"	Datos actualizados
CP13	Dar de baja aula asociada	ID: 2	Mensaje de error por vinculación
CP14	Exportar aulas	Excel	Archivo generado

Tabla 82: Casos de Prueba – Ciclo Académico

CP	Descripción	Entrada	Resultado Esperado
CP15	Registrar nuevo ciclo	"2025 – I"	Registro exitoso
CP16	Registrar ciclo duplicado	"2025 – I"	Error por duplicidad
CP17	Asociar ciclo con alumno	ID ciclo: 1, ID alumno: 10	Asociación correcta
CP18	Listar ciclos	-	Lista con detalles
CP19	Editar ciclo activo	ID: 1, duración: "Mar – Ago"	Cambios guardados
CP20	Dar de baja ciclo vinculado	ID: 2	Error: ciclo vinculado
CP21	Exportar ciclos	Excel	Archivo creado

Tabla 83: Casos de Prueba – Cursos

CP	Descripción	Entrada	Resultado Esperado
CP22	Registrar curso	"Matemáticas Básicas"	Registro exitoso
CP23	Curso duplicado	"Matemáticas Básicas"	Error: curso ya existe
CP24	Asociar curso a estudiante	ID curso: 1, ID estudiante: 6	Asociación correcta
CP25	Listar cursos activos	Filtro: ciclo "2025-I"	Lista mostrada
CP26	Editar curso activo	ID: 1, Duración: "2 meses"	Cambios guardados
CP27	Dar de baja curso vinculado	ID: 2	Error por vinculación
CP28	Exportar cursos	Excel	Exportación exitosa

Tabla 84: Casos de Prueba - Turno

CP	Descripción	Entrada	Resultado Esperado
CP29	Registrar nuevo turno	"Mañana"	Registro exitoso
CP30	Turno duplicado	"Mañana"	Error por duplicidad
CP31	Asociar turno a curso	ID turno: 1, ID curso: 3	Asociación válida
CP32	Listar turnos	-	Lista mostrada con filtros
CP33	Editar turno	ID: 1, nombre: "Tarde"	Datos actualizados
CP34	Baja de turno vinculado	ID: 2	Error por asociación
CP35	Exportar turnos	Excel	Archivo generado

Tabla 85: Casos de Prueba - Matriculas

CP	Descripción	Entrada	Resultado Esperado
CP36	Registrar matrícula	Curso: 1, Estudiante: 5, Ciclo: 1, Turno: 1	Matrícula registrada
CP37	Matrícula duplicada	Mismos datos	Error por duplicación
CP38	Asociar matrícula	-	Asociación exitosa
CP39	Listar matrículas	Filtro: "Estudiante: Juan"	Lista mostrada
CP40	Editar matrícula	ID: 1, turno: 2	Edición exitosa
CP41	Baja de matrícula	ID: 3	Matrícula desactivada
CP42	Exportar matrícula	Excel	Archivo descargado

Tabla 86: Casos de Prueba - Personas

CP	Descripción	Entrada	Resultado Esperado
CP43	Registrar nueva persona	Nombre: Ana, CI: 12345678	Registro exitoso
CP44	Registro duplicado	CI: 12345678	Error: ya registrada
CP45	Clasificar persona	Tipo: Estudiante	Clasificación correcta
CP46	Asignar rol	Rol: Beneficiario	Asignación correcta
CP47	Listar personas	Filtro: "Trabajadores"	Lista mostrada
CP48	Editar persona activa	Cambiar correo	Actualización realizada
CP49	Dar de baja persona	ID: 3	Persona desactivada
CP50	Exportar personas	Excel	Exportación correcta

Tabla 87: Casos de Prueba - Tipo de Identificación

CP	Descripción	Entrada	Resultado Esperado
CP51	Registrar nuevo tipo	"Pasaporte"	Registro exitoso
CP52	Tipo duplicado	"Pasaporte"	Error por duplicidad
CP53	Asociar tipo con persona	ID tipo: 1, ID persona: 5	Asociación correcta
CP54	Listar tipos	-	Lista mostrada
CP55	Editar tipo activo	Cambiar nombre	Actualización exitosa
CP56	Baja de tipo vinculado	ID: 2	Error por asociación
CP57	Exportar tipos	PDF	Archivo exportado

Tabla 88: Casos de Prueba – Asistencia

CP	Descripción	Entrada	Resultado Esperado
CP58	Registrar asistencia	Estudiante: 5, Fecha: hoy	Registro exitoso
CP59	Asistencia duplicada	Mismos datos	Error por duplicado
CP60	Filtrar asistencia	Curso: "Lenguaje", Fecha: hoy	Lista mostrada
CP61	Editar asistencia	Marcar como presente	Registro actualizado
CP62	Exportar asistencia	Excel	Exportación exitosa

Tabla 89: Casos de Prueba - Notas

CP	Descripción	Entrada	Resultado Esperado
CP63	Registrar notas	Curso: 1, Estudiante: 5, Eval: 1-4	Registro correcto
CP64	Notas duplicadas	Mismos datos	Error de duplicidad
CP65	Filtrar notas	Estudiante: 5	Lista mostrada
CP66	Editar nota	Cambiar evaluación 2	Edición exitosa
CP67	Exportar notas	PDF	Archivo generado

Tabla 90: Casos de Prueba - Consultas Médicas y Psicológicas

CP	Descripción	Entrada	Resultado Esperado
CP68	Registrar consulta médica	Nombre: Consulta 1	Registro exitoso
CP69	Consulta duplicada	Mismo nombre	Error
CP70	Asociar consulta	Persona: 6	Asociación correcta
CP71	Listar consultas	Filtro: "Psicológica"	Lista mostrada
CP72	Editar consulta	Cambiar fecha	Actualización correcta
CP73	Exportar consultas	PDF	Archivo generado

- **Organización de las Pruebas**

Cada módulo funcional posee una clase de pruebas específica, la cual contiene métodos decorados con el atributo [TestMethod], que definen escenarios concretos para validar el comportamiento de los servicios. Estas clases están agrupadas dentro de un proyecto de pruebas independiente, referenciado al proyecto principal.

Para simular el comportamiento del sistema sin necesidad de una base de datos real, se emplean repositorios in-memory simulados y objetos de prueba predefinidos.

- **Ejemplo de Código de Prueba Unitaria**

A continuación, se presenta un ejemplo real de una prueba unitaria desarrollada para el módulo de Profesiones. Esta prueba valida el comportamiento del método de registro cuando se intenta guardar una profesión con un nombre duplicado:

Tabla 91: Código de Prueba Unitaria

```
[TestClass]
public class ProfesionServiceTests
{
    private IProfesionService _profesionService;

    [TestInitialize]
    public void Inicializar()
    {
        var mockRepositorio = new Mock<IProfesionRepositorio>();
        mockRepositorio.Setup(r => r.ExisteNombre("Psicólogo")).Returns(true);

        _profesionService = new ProfesionService(mockRepositorio.Object);
    }

    [TestMethod]
    public void RegistrarProfesion_NombreDuplicado_DeberiaFallar()
    {
        var profesionDuplicada = new ProfesionRequestDto
        {
            Nombre = "Psicólogo"
        };

        var resultado = _profesionService.Registrar(profesionDuplicada);

        Assert.IsFalse(resultado.Exito);
        Assert.AreEqual("Ya existe una profesión con ese nombre.", resultado.Mensaje);
    }
}
```

- **Conclusión de las pruebas.**

La implementación de pruebas unitarias mediante MSTest permitió validar el comportamiento de los servicios del sistema de forma confiable y desacoplada del entorno de datos. Este enfoque facilita la identificación temprana de errores, mejora la calidad del código y permite aplicar refactorizaciones con mayor seguridad.

La muestra de código presentada es representativa del estilo y estructura adoptados en la implementación general de pruebas para todos los módulos administrativos. A medida que avance el desarrollo, se seguirán generando pruebas adicionales para alcanzar una mayor cobertura y asegurar la estabilidad del sistema PROYSMAPA en su versión final.

7.12. IMPLANTACIÓN DEL SISTEMA

Una vez culminado el proceso de desarrollo del sistema web y de sus aplicaciones complementarias, se dio inicio a la fase de implantación en el entorno de producción, la cual fue abordada de manera progresiva durante los Sprints 10, 11 y 12. En el Sprint 10 (semanas 19-20), se realizaron ajustes finales y pruebas de aceptación con usuarios clave, lo que permitió validar la funcionalidad y corregir errores detectados. Posteriormente, en el Sprint 11 (semanas 21-22), se llevaron a cabo las actividades necesarias para preparar el sistema para su despliegue, incluyendo la capacitación del personal técnico y administrativo, así como la revisión de parámetros de seguridad y documentación.

Finalmente, en el Sprint 12 (semanas 23-24), se efectuó el despliegue oficial del sistema PROYSMAPA en el entorno productivo, acompañado de un monitoreo inicial para verificar su estabilidad y recolección de retroalimentación por parte de los usuarios. Esta etapa representa un hito fundamental del proyecto, al permitir que las soluciones diseñadas sean finalmente puestas a disposición de los usuarios finales, cumpliendo con los objetivos funcionales y operativos establecidos desde el inicio.

- **Proceso de Implantación**

La implantación se realizó siguiendo una planificación previamente establecida, que incluyó:

La instalación del sistema en los servidores o entornos definitivos. La configuración de bases de datos y servicios asociados.

La validación de conectividad y funcionamiento general en condiciones reales. La preparación del entorno de respaldo (backups automáticos y manuales).

La verificación de usuarios, roles y accesos.

Este proceso fue supervisado por el equipo de desarrollo, en coordinación con el administrador del sistema de la fundación, asegurando una transición fluida desde el entorno de pruebas al de producción.

- **Capacitación y Transferencia**

Un componente clave de la implantación fue la capacitación al personal administrativo y técnico que hará uso del sistema. A través de sesiones prácticas se abordaron los siguientes aspectos:

- Ingreso y administración de registros en los distintos módulos.
- Uso correcto de funcionalidades como exportación de datos, edición y baja lógica.
- Recomendaciones sobre el mantenimiento preventivo y actualizaciones del sistema.
- Procedimientos para la atención de incidencias técnicas.

Estas acciones permitieron no solo garantizar el correcto uso del sistema, sino también fomentar la autonomía operativa de los usuarios frente a la nueva tecnología implementada.

- **Entrega Formal y Soporte**

Como parte del cierre formal del proyecto, durante los Sprints 11 y 12 (semanas 21 a 24) se realizaron las actividades correspondientes a la entrega oficial del sistema PROYSMAPA. Esta etapa incluyó la elaboración y firma de actas de entrega, suscritas por el equipo de desarrollo, el administrador del sistema y los responsables de cada área funcional de la Fundación Sara María Parrales.

Estos documentos reflejan el cumplimiento de los entregables acordados, la validación del funcionamiento del sistema en el entorno productivo.

- La conformidad con las funcionalidades entregadas.
- Carta de capacitación e implementación final del sistema

Ver Cartas en Anexo 14: Carta de Entrega y Soporte de Proyecto

Además, se definió un periodo inicial de soporte post-implantación, en el cual el equipo desarrollador brindará acompañamiento para resolver dudas, aplicar correcciones menores y asegurar la estabilización total del sistema en su nuevo entorno.

VIII. CONCLUSIONES

Como resultado del desarrollo del presente proyecto, se logró implementar exitosamente un sistema web para la gestión administrativa y educativa de la Fundación PROYSMAPA. Esta solución tecnológica abarca múltiples módulos funcionales que permiten gestionar de manera eficiente la información relacionada con personas, profesiones, aulas, ciclos académicos, cursos, turnos, matrículas, asistencia escolar, notas académicas, tipos de identificación y consultas médicas y psicológicas.

El sistema desarrollado responde de forma directa a las necesidades identificadas al inicio del proyecto, facilitando la organización de procesos internos y mejorando significativamente la trazabilidad, control y disponibilidad de la información institucional.

Entre los logros más relevantes del proyecto, se pueden destacar los siguientes:

- Se desarrolló una solución integral bajo el patrón arquitectónico Modelo-Vista-Controlador (MVC) utilizando ASP.NET Core 8, lo que garantiza un código estructurado, escalable y mantenible.
- Se aplicaron buenas prácticas de codificación y principios sólidos de ingeniería de software, lo cual asegura que otros desarrolladores puedan comprender y dar mantenimiento al sistema sin mayores dificultades.
- Se implementaron funcionalidades clave como validaciones, exportación de datos, filtros avanzados, control de acceso y baja lógica, que fortalecen el sistema desde el punto de vista operativo y de seguridad.
- Se diseñó una interfaz intuitiva y funcional orientada a facilitar el trabajo del personal administrativo de la fundación.
- Se capacitó al equipo usuario y se realizaron entregas formales con sus respectivas actas, lo cual garantiza la adopción efectiva del sistema en el entorno institucional.

El sistema PROYSMAPA no solo cumple con los objetivos planteados inicialmente,

sino que también representa una herramienta robusta que potenciará la gestión y el impacto social de la fundación. Este proyecto sienta las bases para futuras mejoras y ampliaciones que podrán seguir desarrollándose según las nuevas necesidades que puedan surgir.

IX. RECOMENDACIONES

A partir del proceso de análisis, diseño, desarrollo e implementación del sistema **PROYSMAPA**, se derivan una serie de recomendaciones que permitirán asegurar su sostenibilidad, eficiencia y evolución a lo largo del tiempo. Estas sugerencias están orientadas tanto a los responsables de la administración del sistema como a los futuros desarrolladores que pudieran continuar con su mantenimiento o expansión.

- **Mantenimiento Correctivo y Evolutivo**

Se recomienda establecer un plan de mantenimiento periódico, que permita identificar posibles errores, inconsistencias o mejoras funcionales derivadas del uso cotidiano del sistema.

Es importante documentar cada ajuste o mejora realizada, con el fin de mantener un historial de versiones y cambios aplicados.

- **Capacitación Continua del Personal**

A fin de garantizar el uso correcto de la herramienta, se sugiere brindar capacitaciones periódicas al personal administrativo sobre el uso de cada módulo del sistema.

La creación de un manual de usuario actualizado puede ser de gran utilidad para el aprendizaje de nuevos colaboradores y la resolución de dudas frecuentes.

- **Seguridad y Respaldo de Información**

Se recomienda implementar mecanismos de respaldo automático de la base de datos, con una frecuencia adecuada según el volumen de operaciones realizadas. Es fundamental asegurar que los usuarios tengan niveles de acceso controlados, de acuerdo con su rol dentro de la fundación, para proteger la integridad de la información.

- **Escalabilidad del Sistema**

El sistema fue desarrollado con una arquitectura escalable, por lo cual se recomienda considerar futuras ampliaciones, tales como:

Integración con sistemas de mensajería o notificaciones (SMS, correo electrónico).

Incorporación de reportes gráficos o tableros de control (dashboards).

Desarrollo de una versión móvil para facilitar la gestión en campo.

- **Evaluación de Impacto**

Se sugiere realizar evaluaciones periódicas del impacto del sistema en los procesos de la fundación, lo que permitirá medir mejoras, identificar nuevas necesidades y justificar futuras inversiones tecnológicas.

Estas recomendaciones están orientadas a preservar el valor del trabajo realizado y a fomentar el crecimiento continuo del sistema PROYSMAPA, con una visión clara de mejora continua y adaptación tecnológica al servicio de la misión de la fundación.

X. BIBLIOGRAFIA

- X. Barrera, M. I., & Santa Cruz, S. (2015). Systematic Review of Beneficiary Management Systems. *Journal of Information Systems*, 29(1), 23–34.
- XI. Beck, K. (2000). *Extreme Programming Explained: Embrace Change*. Addison-Wesley.
- XII. Booch, G., Rumbaugh, J., & Jacobson, I. (1999). *The Unified Modeling Language User Guide*. Addison-Wesley.
- XIII. Brundage, D. H., & MacKeracher, D. (1980). *The Foundations of Adult Education in Canada*. OISE Press.
- XIV. Cattell, R. (2018). *JSON at Work: Practical Data Interchange for the Web*. Manning Publications.
- XV. Crockford, D. (2006). *JavaScript Object Notation (JSON)*. Retrieved from <https://www.json.org/>
- XVI. Crockford, D. (2008). *JavaScript: The Good Parts* (1st ed.). O'Reilly Media.
- XIX. Duckett, J. (2014). *HTML and CSS: Design and Build Websites*. Wiley.
- XX. Esposito, D. (2023). *Pro ASP.NET Core 8: MVC, Blazor, and Beyond*. Apress.
- XXI. Flanagan, D. (2020). *JavaScript: The Definitive Guide* (7th ed.). O'Reilly Media.
- XXII. Fowler, M. (2004). *UML Distilled: A Brief Guide to the Standard Object Modeling Language* (3rd ed.). Addison-Wesley.
- XXIII. Freeman, A. (2018). *ASP.NET Core in Action*. Manning Publications.
- XXIV. Freeman, A., & Sanderson, B. (2019). *Pro ASP.NET Core 3: Develop Cloud-Ready Web Applications Using ASP.NET Core* (6th ed.). Apress.
- XXV. Gardner, H. (2011). *Frames of Mind: The Theory of Multiple Intelligences* (3rd ed.). Basic Books.
- XXVI. Highsmith, J. (2002). *Agile Software Development Ecosystems*. Addison-Wesley.
- XXVII. Keith, J. (2011). *Ajax: The Definitive Guide*. O'Reilly Media.

XXVIII. Keith, J. (2013a). *HTML5 for Web Designers*. A Book Apart.

XXIX. Keith, J. (2013b). *CSS3 for Web Designers*. A Book Apart.

Kurose, J. F., & Ross, K. W. (2020). *Computer Networking: A Top-Down Approach* (8th ed.). Pearson.

XXX. Meyer, E. (2011). *CSS3: The Missing Manual*. O'Reilly Media.

XXXI. Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum.org. Retrieved from <https://scrumguides.org/>

XXXII. Steinberg, A., & Lichtenberg, L. (2018). *Pro ASP.NET Core MVC 2* (2nd ed.). Apress.

XXXIII. Tanenbaum, A. S., & Wetherall, D. J. (2011). *Computer Networks* (5th ed.). Pearson.

XXXIV. Williams, A. (2017). *Effective Beneficiary Management Systems: Concepts and Practices*. Cambridge University Press.

XXXV. ZeOldman, J., & Marcotte, E. (2010). *Designing with Web Standards* (3rd ed.). New Riders.

López, M. (2020). *Análisis de la gestión administrativa en organizaciones sin fines de lucro en Nicaragua: el caso de la Asociación Educativa "Mi Futuro"*. [Tesis de posgrado, Universidad Politécnica de Nicaragua]. Repositorio Institucional UPOLI.

XI. ANEXOS

Se adjunta un disco compacto (CD) que contiene los recursos digitales esenciales para la revisión técnica, la implementación y el estudio detallado del proyecto PROYSMAPA. Este repositorio facilita la comprensión y validación del trabajo realizado, así como su posible continuidad en fases futuras.

El contenido del CD incluye:

- Código fuente completo del sistema **PROYSMAPA**, desarrollado en ASP.NET Core 8 y estructurado bajo el patrón arquitectónico Modelo-Vista-Controlador (MVC).
- Documento digital de la tesis, en formato PDF, correspondiente al desarrollo íntegro del proyecto monográfico.
- Presentación del proyecto, utilizada en la defensa del trabajo, en formato editable y PDF.
- Resultados de los Sprint

Se recomienda respaldar este contenido en medios de almacenamiento alternos y mantenerlo disponible para consulta o reutilización conforme se desarrollen nuevas versiones del sistema.

SOLUCION A LOS SPRINT

Sprint 1 – Análisis de Requerimientos y Diseño Inicial

Anexo 1: Análisis de Requerimientos y Diseño Inicial

Durante el Sprint 1 se sentaron las bases del proyecto mediante un análisis exhaustivo de los requerimientos funcionales y no funcionales del sistema PROYSMAPA. Este proceso incluyó reuniones estructuradas con los stakeholders: directores, docentes y personal administrativo, donde se levantaron necesidades clave como la gestión académica, control de asistencia, evaluaciones, y seguimiento médico y psicológico. A partir de estas sesiones, se elaboró el Product Backlog inicial, priorizando funcionalidades esenciales para el funcionamiento del sistema.

ID	Product BackLog	Prioridad	Módulo	Criterios de Aceptación
PB01	Como administrador, quiero gestionar las profesiones para asignarlas a las personas.	Alta	Profesiones	Puede crear, editar, eliminar y listar profesiones activas e inactivas.
PB02	Como administrador, quiero registrar aulas para asignarlas a grados y turnos.	Alta	Aulas	Puede agregar aulas, asociarlas a grados y ver disponibilidad.
PB03	Como usuario, quiero iniciar sesión en el sistema con mi	Alta	Autenticación	Login funcional, validación de credenciales, mensajes de error claros.

	usuario y contraseña.			
PB04	Como administrador, quiero definir ciclos académicos para organizar la matrícula.	Alta	Ciclos Académicos	Crear ciclos activos con fecha de inicio y fin.
PB05	Como administrador, quiero crear y asignar cursos por grado.	Alta	Cursos	Asignar cursos a grados, evitar duplicaciones y gestionar estado activo.
PB06	Como psicólogo, quiero registrar consultas psicológicas asociadas a un estudiante.	Media	Consultas Psicológicas	Crear y consultar registros, vincular a estudiantes y fechas.
PB07	Como médico, quiero registrar consultas médicas por estudiante.	Media	Consultas Médicas	Ingresar signos vitales, motivo de consulta y diagnóstico clínico.
PB08	Como administrativo, quiero matricular estudiantes en un ciclo académico.	Alta	Matrícula	Matricular alumnos por aula, grado y turno en un ciclo activo.

PB09	Como docente, quiero registrar asistencia diaria de mis estudiantes.	Alta	Asistencia	Ver lista de estudiantes por curso, marcar presentes/ausentes.
PB10	Como director, quiero generar reportes de asistencia y rendimiento académico.	Media	Reportes	Consultas por fechas, por aula, por estudiante y exportación a PDF o Excel.
PB11	Como administrador, quiero asignar roles a los usuarios del sistema.	Alta	Seguridad / Roles	Asignar uno o varios roles por usuario, controlar permisos sobre los módulos.
PB12	Como usuario, quiero tener una interfaz fácil de usar y adaptable a dispositivos.	Alta	Experiencia de Usuario	Diseño responsivo, tiempos de carga aceptables, validaciones claras.
PB13	Como administrador, quiero auditar acciones importantes del sistema.	Media	Auditoría	Registrar cambios de configuración, logins, y actividades críticas.

En este primer sprint se sentaron las bases para el desarrollo del sistema. Se realizó un análisis de los requerimientos funcionales y no funcionales en colaboración con los stakeholders, recopilando información clave para definir el alcance del proyecto.

Este sprint permitió al equipo alinear expectativas y definir una visión clara del producto.

Sprint 2 – Configuración del Entorno y Arquitectura del Sistema

Anexo 2: Configuración del Entorno y Arquitectura del Sistema

En el segundo sprint se logró configurar completamente el entorno de desarrollo, incluyendo la instalación de Visual Studio, SQL Server, Git, y herramientas de colaboración como Azure DevOps. Se definió la arquitectura basada en ASP.NET Core 8, utilizando el patrón MVC para una separación clara de responsabilidades. Se creó una solución modular que incluyó proyectos para la lógica de negocio, acceso a datos y pruebas. Además, se diseñaron y construyeron los primeros diagramas, incluyendo casos de uso, diagramas de clases, los cuales fueron validados con los stakeholders. Se implementó la base de datos inicial.

Tarea	Categoría	Asignado a	Estimación de tiempo
Instalación y configuración de Visual Studio 2022	Configuración del entorno de desarrollo	Nestor A.	2h
Instalación y configuración de SQL Server y SSMS	Configuración del entorno de desarrollo	Samuel M.	2h
Configuración del repositorio Git	Configuración del entorno de desarrollo	Nestor A.	1h

Creación y vinculación del proyecto en Azure DevOps	Configuración del entorno de desarrollo	Nestor A.	1.5h
Definición de reglas de colaboración (ramas, PRs, commits)	Configuración del entorno de desarrollo	Samuel M.	1h
Definición de arquitectura MVC con ASP.NET Core 8	Arquitectura del sistema	Samuel M.	2h
Creación de solución modular con capas (WebApp, Application, Domain, Infrastructure, Tests)	Arquitectura del sistema	Samuel M.	3h
Configuración de inyección de dependencias entre capas	Arquitectura del sistema	Nestor A.	1.5h
Establecimiento de convenciones	Arquitectura del sistema	Nestor A.	1h

de carpetas y nombres			
Elaboración de diagramas de casos de uso	Diseño UML y validación	Samuel M.	2h
Creación de diagramas de clases	Diseño UML y validación	Nestor A.	2h
Revisión y validación de diagramas UML con stakeholders	Diseño UML y validación	Samuel M.	1.5h
Ajuste de diagramas UML según retroalimentación	Diseño UML y validación	Samuel M.	1.5h
Diseño del modelo entidad-relación	Base de datos inicial	Nestor A.	2h
Creación de estructura base de la base de datos en SQL Server	Base de datos inicial	Nestor A.	2h
Generación de scripts para	Base de datos inicial	Samuel M.	1.5h

tablas y relaciones			
Pruebas de conexión entre BD y acceso a datos	Base de datos inicial	Samuel M.	1h
Documentación de la estructura de base de datos	Base de datos inicial	Samuel M.	1h

En este sprint se consolidó la arquitectura técnica del sistema y se configuró el entorno de desarrollo. Se implementó la estructura de la base de datos en SQL Server, definiendo tablas, relaciones y restricciones iniciales a partir del análisis previo.

Sprint 3 – Desarrollo Inicial de Módulos e Interfaces

Anexo 3: Desarrollo Inicial de Módulos e Interfaces

Durante este sprint se comenzó con la codificación de los primeros módulos funcionales, destacando la gestión de usuarios, profesiones y aulas. Se desarrollaron formularios utilizando Razor Pages, validaciones del lado cliente con jQuery y del lado servidor con anotaciones de datos en C#. Se conectaron estas interfaces con la base de datos usando Entity Framework Core, logrando la persistencia de información en formularios CRUD. Se implementó un sistema de login básico utilizando autenticación por cookies. Además, se diseñaron las primeras vistas parciales reutilizables para encabezados, menús y alertas, lo cual permitió estandarizar la interfaz. Las funcionalidades fueron organizadas en áreas por módulo, facilitando su mantenimiento. Finalmente, se estableció el proceso de despliegue local y publicación en un servidor de pruebas interno.

Tarea	Categoría	Asignado a	Estimación de tiempo
Desarrollo del módulo de gestión de usuarios (CRUD)	Módulos funcionales	Nestor A.	3h
Desarrollo del módulo de gestión de profesiones (CRUD)	Módulos funcionales	Nestor A.	2.5h
Desarrollo del módulo de gestión de aulas (CRUD)	Módulos funcionales	Nestor A.	2.5h
Implementación de formularios Razor Pages para los módulos	Interfaces de usuario	Samuel M.	2h
Aplicación de validaciones del lado cliente con jQuery	Interfaces de usuario	Samuel M.	1.5h
Aplicación de validaciones del lado servidor con	Interfaces de usuario	Samuel M.	1.5h

anotaciones de datos en C#			
Configuración de Entity Framework Core para acceso a datos	Persistencia de datos	Nestor A.	1.5h
Implementación de operaciones CRUD con EF Core	Persistencia de datos	Nestor A.	2.5h
Desarrollo de sistema de login básico con autenticación por cookies	Autenticación y seguridad	Nestor A.	2h
Creación de vistas parciales para encabezados reutilizables	Interfaz gráfica	Samuel M.	1h
Creación de vistas parciales para menús y alertas reutilizables	Interfaz gráfica	Samuel M.	1.5h
Organización de funcionalidades	Arquitectura y mantenimiento	Nestor A.	1h

en áreas por módulo			
Configuración del proceso de despliegue local	Despliegue	Nestor A.	1.5h
Publicación en servidor de pruebas interno	Despliegue	Samuel M.	1h

Durante este sprint se desarrollaron los primeros módulos funcionales del sistema, enfocados en la gestión de usuarios y actividades, así como en la creación de las primeras interfaces gráficas básicas. Se configuraron herramientas y librerías necesarias para optimizar la programación y se implementaron las primeras interacciones entre la interfaz y la base de datos.

Sprint 4 – Implementación de Funcionalidades Clave (Matricula y Vida Estudiantil)

Anexo 4: Implementación de Funcionalidades Clave

Este sprint se centró en la implementación de funcionalidades fundamentales para el sistema académico. Se integraron los módulos de Matricula y Vida Estudiantil, permitiendo registrar y gestionar elementos esenciales del contexto educativo. Se incorporó la lógica de negocio asociada a cada uno.

También se aplicó refactorización de código, con énfasis en mejorar la reutilización y la legibilidad. Este sprint marcó el inicio formal de las buenas prácticas de aseguramiento de calidad.

Tarea	Categoría	Asignado a	Estimación de tiempo
Desarrollo del módulo de	Módulos académicos	Samuel M.	3h

Matrícula (registro y edición)			
Desarrollo del módulo de Vida Estudiantil (gestión de historial y eventos)	Módulos académicos	Samuel M.	2.5h
Implementación de lógica de negocio para Matrícula	Lógica de negocio	Nestor A.	1.5h
Implementación de lógica de negocio para Vida Estudiantil	Lógica de negocio	Nestor A.	1.5h
Refactorización de código de formularios y controladores	Refactorización y mantenimiento	Samuel M.	2h
Reorganización de funciones repetidas en métodos reutilizables	Refactorización y mantenimiento	Samuel M.	1.5h

Mejora de legibilidad y limpieza del código (nombres, comentarios, estructura)	Refactorización y mantenimiento	Nestor A.	1.5h
Aplicación de principios SOLID donde corresponde	Buenas prácticas de desarrollo	Nestor A.	2h
Documentación inicial de criterios de calidad de código	Aseguramiento de calidad	Samuel M.	1h
Establecimiento de convenciones de codificación en el equipo	Aseguramiento de calidad	Samuel M.	1h

Este sprint se centró en la implementación de funcionalidades fundamentales para el sistema académico. Se integraron los módulos de Matricula y Vida Estudiantil, permitiendo registrar y gestionar elementos esenciales del contexto educativo. Se incorporó la lógica de negocio asociada a cada uno.

También se aplicó refactorización de código, con énfasis en mejorar la reutilización y la legibilidad. Este sprint marcó el inicio formal de las buenas prácticas de aseguramiento de calidad.

Sprint 5 – Funcionalidades Adicionales y Revisión del Diseño

Anexo 5: Funcionalidades Adicionales y Revisión del Diseño

Durante este sprint se expandió la cobertura funcional del sistema con nuevos módulos como Asistencias. También se incorporaron filtros y búsquedas con select2 para facilitar la interacción con grandes volúmenes de datos. Se revisó en profundidad el diseño visual de las pantallas, implementando mejoras en usabilidad, agrupación de campos, distribución en pestañas y estilos personalizados. Se validó el cumplimiento de estándares de accesibilidad.

Tarea	Categoría	Asignado a	Estimación de tiempo
Desarrollo del módulo de Asistencias (registro, listado, edición)	Módulos adicionales	Samuel M.	3h
Implementación de filtros dinámicos con Select2	Mejora de interacción	Samuel M.	2h
Implementación de búsquedas avanzadas con Select2 para campos con gran volumen	Mejora de interacción	Samuel M.	2h
Revisión del diseño visual de	Usabilidad y diseño	Nestor A.	1.5h

las pantallas principales			
Agrupación de campos relacionados en formularios	Usabilidad y diseño	Nestor A.	1h
Organización de contenidos en pestañas (tabs)	Usabilidad y diseño	Nestor A.	1h
Aplicación de estilos personalizados a formularios y componentes	Usabilidad y diseño	Samuel M.	2h
Validación del cumplimiento de estándares de accesibilidad (WCAG)	Accesibilidad	Samuel M.	1.5h
Ajuste de etiquetas, colores y elementos de contraste	Accesibilidad	Samuel M.	1h

En este sprint se ampliaron las funcionalidades del sistema con el desarrollo de módulos adicionales y la incorporación de lógica de negocio complementaria. Se revisó el diseño de las interfaces y la estructura de navegación, ajustándolos según la retroalimentación obtenida de las pruebas internas. Este proceso permitió mejorar

la coherencia visual, optimizar la experiencia del usuario y garantizar que las nuevas funcionalidades se integraran de forma fluida con los módulos existentes.

Sprint 6 – Módulos de Usuario y Seguridad

Anexo 6: Módulos de Usuario y Seguridad

Este sprint permitió robustecer la seguridad del sistema mediante la implementación de un sistema de autenticación completo. Se creó el módulo de gestión de usuarios con funcionalidades como registro, edición y eliminación lógica. Se configuró la autenticación basada en roles, integrando lógica para limitar el acceso a ciertas vistas según permisos. También se integraron validaciones personalizadas y manejo de errores centralizado. Se desarrollaron vistas para administración de perfiles y se vinculó cada usuario a un conjunto específico de permisos.

Tarea	Categoría	Asignado a	Estimación de tiempo
Desarrollo del módulo de gestión de usuarios (registro, edición, eliminación lógica)	Gestión de usuarios	Nestor A.	3h
Implementación del sistema de autenticación completo con cookies y roles	Autenticación y seguridad	Nestor A.	2.5h
Configuración de autorización por roles para limitar acceso a vistas	Autenticación y seguridad	Samuel M.	2h

Desarrollo de vistas para administración de perfiles de usuario	Gestión de usuarios	Samuel M.	2h
Asociación de cada usuario a un conjunto de permisos definidos	Seguridad y control de acceso	Samuel M.	1.5h
Implementación de validaciones personalizadas en formularios	Validación de datos	Nestor A.	1.5h
Desarrollo de manejo de errores centralizado (filtros/middleware)	Control de errores	Nestor A.	2h
Revisión de rutas y protección contra acceso no autorizado	Seguridad	Samuel M.	1h

Durante este sprint se implementaron los módulos de gestión de usuarios y seguridad, incluyendo autenticación y administración de perfiles. Se establecieron mecanismos de control de acceso y validaciones para proteger la información del sistema.

Sprint 7 – Finalización de Funcionalidades de Usuario y Pruebas de Usabilidad

Anexo 7: Finalización de Funcionalidades de Usuario y Pruebas de Usabilidad

En este sprint se dio cierre a las funcionalidades del área de usuarios, enfocándose en refinar la experiencia del usuario final. Se mejoraron las transiciones entre vistas, se ajustaron los mensajes de retroalimentación visual, y se incorporaron animaciones sutiles para reforzar la interacción. Se fortaleció la gestión de sesiones, se implementaron alertas personalizadas con SweetAlert y se realizaron mejoras en la navegabilidad del menú principal.

Tarea	Categoría	Asignado a	Estimación de tiempo
Refinamiento final del módulo de gestión de usuarios	Gestión de usuarios	Nestor A.	2h
Mejora de transiciones entre vistas (navegación fluida)	Experiencia de usuario	Samuel M.	1.5h
Ajuste de mensajes de retroalimentación visual (mensajes de éxito, error, validación)	Experiencia de usuario	Samuel M.	1.5h
Incorporación de animaciones	Experiencia de usuario	Nestor A.	1.5h

sutiles para botones, formularios y paneles			
Fortalecimiento de la gestión de sesiones (timeout, cierre seguro)	Seguridad y sesiones	Samuel M.	2h
Implementación de alertas personalizadas con SweetAlert	Interfaz y retroalimentación	Samuel M.	1.5h
Mejora de la estructura y navegabilidad del menú principal	Navegación	Nestor A.	2h
Pruebas internas de usabilidad y recolección de retroalimentación	Pruebas de usabilidad	Samuel M.	2h

En este sprint se finalizó el desarrollo de las funcionalidades relacionadas con los usuarios, ajustando y refinando las interfaces para mejorar la accesibilidad y facilidad de uso. Se ejecutaron pruebas de usabilidad con usuarios representativos, lo que permitió identificar y corregir áreas de mejora. Este trabajo garantizó que el sistema ofreciera una experiencia intuitiva, adaptada a las necesidades de su público objetivo.

Sprint 8 – Funcionalidades Avanzadas y Pruebas del Sistema

Anexo 8: Funcionalidades Avanzadas y Pruebas del Sistema

Este sprint abordó la creación de funcionalidades avanzadas como reportes, exportación de datos en Excel. Se desarrollaron consultas SQL optimizadas para alimentar estos reportes. También se construyeron servicios AJAX que permitieron refrescar secciones de la pantalla sin recargar toda la página, mejorando la interactividad.

Tarea	Categoría	Asignado a	Estimación de tiempo
Desarrollo de funcionalidad de generación de reportes (tablas y métricas)	Funcionalidades avanzadas	Samuel M.	2.5h
Implementación de exportación de datos a Excel (formato y descarga)	Funcionalidades avanzadas	Samuel M.	2h
Escritura de consultas SQL optimizadas para reportes	Acceso a datos	Nestor A.	2h
Validación de rendimiento y tiempos de	Pruebas de rendimiento	Samuel M.	1.5h

respuesta de reportes			
Desarrollo de servicios AJAX para refrescar secciones específicas	Interactividad y rendimiento	Nestor A.	2h
Integración de servicios AJAX con formularios y componentes dinámicos	Interactividad y rendimiento	Samuel M.	2h
Pruebas del sistema completas sobre módulos con nuevas funcionalidades	Pruebas del sistema	Samuel M.	2h

Este sprint se enfocó en la incorporación de funcionalidades avanzadas como la generación de reportes y el uso de AJAX para optimizar la interacción dinámica entre el cliente y el servidor

Sprint 9 – Intercambio de Datos con JSON

Anexo 9: Intercambio de Datos con JSON

Se aplicaron políticas de serialización personalizada y manejo de errores estructurado. Asimismo, se integraron servicios que permitieron la consulta y actualización de datos sin necesidad de recargar páginas, usando llamadas

asincrónicas.

Tarea	Categoría	Asignado a	Estimación de tiempo
Implementación de políticas de serialización personalizada (nombres, formatos, fechas)	Intercambio de datos	Nestor A.	2h
Desarrollo de manejo de errores estructurado en respuestas JSON	Control de errores	Samuel M.	1.5h
Integración de servicios JSON para consulta asincrónica de datos	Servicios asincrónicos	Samuel M.	2h
Integración de servicios JSON para actualización asincrónica de datos	Servicios asincrónicos	Nestor A.	2h

Validación del correcto retorno de códigos de estado HTTP y mensajes	Interoperabilidad	Nestor A.	1.5h
Pruebas de consumo de servicios JSON en el frontend (AJAX, JS)	Pruebas y validación	Samuel M.	1.5h

En este sprint se integró el intercambio de datos mediante JSON, lo que mejoró significativamente la eficiencia y la interoperabilidad del sistema con otros módulos y servicios. Se validó la comunicación interna entre componentes y se realizaron pruebas de integración final para asegurar la correcta transmisión de información, optimizando los tiempos de respuesta y reduciendo la carga de procesamiento.

Sprint 10 – Ajustes Finales y Pruebas de Aceptación

Anexo 10: Ajustes Finales y Pruebas de Aceptación

Este sprint fue dedicado a consolidar el sistema de cara a la entrega final. Se corrigieron algunos errores pendientes visualizados en los sprint anteriores. Se realizaron pruebas de aceptación con usuarios clave del centro educativo, quienes validaron cada módulo según los criterios definidos. Se generó una versión candidata para producción que integró todos los módulos funcionales bajo un diseño coherente. Este sprint sirvió como validación final del sistema.

Tarea	Categoría	Asignado a	Estimación de tiempo
-------	-----------	------------	----------------------

Corrección de errores visuales y funcionales identificados en sprints anteriores	Corrección de errores	Nestor A.	2h
Revisión general de módulos funcionales para integración final	Consolidación del sistema	Samuel M.	1.5h
Ejecución de pruebas de aceptación con usuarios clave	Pruebas de aceptación	Nestor A.	2h
Ajuste de funcionalidades basado en retroalimentación de aceptación	Validación de usuario	Nestor A.	1.5h
Generación de versión candidata para producción	Entregables finales	Nestor A.	1.5h
Verificación del diseño coherente y navegación entre módulos	Revisión de diseño	Samuel M.	1.5h
Validación final del sistema como	Cierre de sprint	Samuel M.	1h

conjunto integrado			
-----------------------	--	--	--

Este sprint se dedicó a la corrección de errores y a la realización de pruebas de aceptación con usuarios finales, garantizando que el sistema cumpliera con los criterios de calidad establecidos. Se revisó y actualizó la documentación técnica y funcional, asegurando que estuviera alineada con la versión final del producto antes de su despliegue.

Sprint 11 – Preparación para el Despliegue

Anexo 11: Preparación para el Despliegue

Durante este sprint se completaron los preparativos necesarios para el paso del sistema a un entorno productivo. Además, se capacitó al personal administrativo y técnico sobre el uso de las funcionalidades, gestión de usuarios, generación de reportes y resolución de problemas comunes. Se afinó el sistema de respaldos automáticos y se ajustaron parámetros de seguridad y auditoría.

Tarea	Categoría	Asignado a	Estimación de tiempo
Configuración del entorno productivo (base de datos, servidor, rutas)	Despliegue	Nestor A.	2h
Revisión de parámetros de seguridad y	Seguridad y auditoría	Samuel M.	1.5h

políticas de auditoría			
Ajuste final del sistema de respaldos automáticos	Respaldo y mantenimiento	Nestor A.	1.5h
Elaboración de guía rápida de uso para el personal	Capacitación y soporte	Nestor A.	1.5h
Capacitación al personal administrativo sobre uso del sistema	Capacitación	Samuel M.	2h
Capacitación al personal técnico sobre mantenimiento y solución de problemas	Capacitación	Samuel M.	2h
Simulación de incidentes comunes para validación de soporte	Validación operativa	Nestor A.	1h

Durante este sprint se completaron todos los preparativos necesarios para el despliegue en el entorno de producción. Se brindó capacitación al personal administrativo y técnico en el uso y administración del sistema, incluyendo la gestión de usuarios, la generación de reportes y la solución de problemas comunes. Asimismo, se implementaron ajustes finales derivados de la retroalimentación de los usuarios, reforzando la seguridad y la estabilidad del sistema.

Sprint 12 – Despliegue en Producción y Monitoreo

Anexo 12: Despliegue en Producción y Monitoreo

En el sprint final se llevó a cabo la implementación oficial del sistema PROYSMAPA en el entorno de producción. Se validó su funcionamiento en condiciones reales y se realizó un monitoreo constante durante los primeros días, a través de registros de errores y seguimiento de uso. Este sprint marcó el cierre exitoso de la fase de desarrollo, dejando al sistema funcionando plenamente y al equipo preparado para su mantenimiento y mejora continua.

Tarea	Categoría	Asignado a	Estimación de tiempo
Implementación oficial del sistema en entorno de producción	Despliegue final	Samuel M.	2h
Validación del funcionamiento en condiciones reales	Verificación en producción	Nestor A.	1.5h
Configuración de registros de errores y bitácoras de uso	Monitoreo	Samuel M.	1.5h

Monitoreo del sistema en los primeros días de operación	Seguimiento post-despliegue	Nestor A.	2h
Resolución de errores menores detectados en producción	Soporte inmediato	Nestor A.	1.5h
Reunión de cierre con el equipo de desarrollo	Cierre de proyecto		1h
Entrega formal del sistema y documentación final	Cierre de proyecto	Samuel M.	1.5h

En el sprint final se realizó el despliegue oficial del sistema PROYSMAPA en el entorno de producción, asegurando su correcto funcionamiento en condiciones reales. Se ejecutó un monitoreo constante durante los primeros días para detectar y resolver posibles incidencias. La retroalimentación obtenida permitió identificar mejoras a implementar en futuras versiones. Este sprint marcó el cierre exitoso del ciclo de desarrollo y el inicio de la etapa de mantenimiento continuo.

ENTREVISTA

Anexo 13: Entrevista a PROYSMAPA

A continuación, se presenta una recopilación de preguntas realizadas durante el proceso de entrevistas a los distintos actores del proyecto PROYSMAPA. Estas fueron fundamentales para levantar los requerimientos funcionales y técnicos del sistema, garantizando que la solución desarrollada respondiera a las necesidades reales de los usuarios.

1. Datos generales y necesidades institucionales

- ¿Qué tipo de información consideran fundamental registrar sobre cada estudiante?
- ¿Qué datos personales y académicos se deben almacenar para los docentes y voluntarios?
- ¿Cómo identifican actualmente a cada persona (documento, código, cédula, etc.)?
- ¿Qué procesos administrativos se realizan con mayor frecuencia?

2. Gestión académica y ciclos escolares

- ¿Cómo organizan los ciclos académicos (inicio, fin, nombre, año)?
- ¿Con qué frecuencia se crean o modifican cursos y grados?
- ¿Cómo se asignan los docentes a los cursos?
- ¿Qué estructura académica utilizan (grados, aulas, turnos, secciones)?
- ¿Cuántos estudiantes promedio hay por aula y turno?

3. Matrícula y seguimiento de estudiantes

- ¿Cómo realizan el proceso de matrícula actualmente?
- ¿Qué información se debe guardar en cada matrícula?
- ¿Tienen estudiantes que se matriculan en más de un curso o programa?
- ¿Es importante llevar un historial de matrículas pasadas?

4. Control de asistencia

- ¿Cómo registran la asistencia de los estudiantes diariamente?
- ¿Desean justificar ausencias? ¿Quién lo debe hacer?
- ¿Necesitan reportes de asistencia por aula, turno o estudiante?

- ¿Qué nivel de detalle desean en los registros de asistencia (presente, ausente, justificado, etc.)?

5. Notas y desempeño

- ¿Cómo se gestionan actualmente las calificaciones?
- ¿Qué tipo de evaluaciones aplican (exámenes, tareas, comportamiento, etc.)?
- ¿Desean generar boletas de notas? ¿Con qué frecuencia?
- ¿Qué reglas deben cumplirse para aprobar un curso o grado?

6. Consultas médicas y psicológicas

- ¿Qué datos deben registrarse durante una consulta médica o psicológica?
- ¿Quién puede acceder a esta información? ¿Desean mantenerla confidencial?
- ¿Se requiere llevar un historial de consultas por estudiante?

7. Gestión de usuarios y permisos

- ¿Qué tipos de usuarios deben tener acceso al sistema (administrador, docente, voluntario, médico)?
- ¿Qué acciones debe poder realizar cada tipo de usuario?
- ¿Desean implementar niveles de acceso por roles o perfiles?

8. Reportes y estadísticas

- ¿Qué tipos de reportes utilizan actualmente o desean generar (asistencia, matrícula, notas, consultas)?
- ¿Con qué frecuencia necesitan estos reportes? ¿En qué formatos?
- ¿Desean acceder a estadísticas gráficas o indicadores visuales en algún panel?

