UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA, MANAGUA
UNAN-MANAGUA

2026: "Por una Educación al Servicio del Pueblo, ¡Viva Sandino!"

DEPARTAMENTO DE CIENCIAS TECNOLÓGICAS Y SALUD

INGENIERÍA EN SISTEMAS DE INFORMACIÓN

I SEMESTRE

II Año

" Desarrollo de un sistema de escritorio para el Instituto Nacional 21 de Junio "

Grupo 4

AUTORES:

- Nerybeth Auxiliadora Bravo López
- Anderson Johan Gonzalez Torrez
- Pablo Francisco Bravo Centeno
- Yeison Alexander Gomez Morales

DOCENTES

- José Jonathan Moreno Castro
- Yesenia Sobeyda Téllez Gomez
- María Jorlenis Ríos Pérez
- Isamar Del Socorro Bonilla Martinez
- Carlos Miguel Matamoros Osorio

Mayo, 2026

UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA, MANAGUA
UNAN-MANAGUA

2026: "Por una Educación al Servicio del Pueblo, ¡Viva Sandino!"

DEPARTAMENTO DE CIENCIAS TECNOLÓGICAS Y SALUD

INGENIERÍA EN SISTEMAS DE INFORMACIÓN

I SEMESTRE

II Año

" Desarrollo de un sistema de escritorio para el Instituto Nacional 21 de Junio "

Grupo 4

AUTORES:

- Nerybeth Auxiliadora Bravo López (0009-0004-8994-7504)
- Anderson Johan Gonzalez Torrez (0009-0005-7522-8303)
- Pablo Francisco Bravo Centeno (0009-0003-4336-041X)
- Yeison Alexander Gomez Morales (0009-0009-2135-4207)

DOCENTES

- José Jonathan Moreno Castro
- Yesenia Sobeyda Téllez Gomez
- María Jorlenis Ríos Pérez
- Isamar Del Socorro Bonilla Martinez
- Carlos Miguel Matamoros Osorio

Mayo, 2026

RESUMEN

El presente proyecto aborda el desarrollo de un sistema automatizado de escritorio para el registro y monitoreo de hábitos saludables en estudiantes del Instituto Nacional 21 de Junio. Actualmente, la institución realiza el seguimiento de hábitos de los estudiantes de manera manual, lo que genera pérdida de información, dificultad para consultar datos históricos y ausencia de reportes automatizados. La solución propuesta utiliza el paradigma de Programación Orientada a Objetos (POO), una base de datos relacional para almacenar registros diarios y módulos de estadística descriptiva para calcular promedios de cumplimiento. El sistema permitirá a docentes registrar hábitos diariamente, a Administradores generar reportes en PDF y a visualizar indicadores clave para la toma de decisiones. El proyecto se enmarca en la línea de investigación CED-1: Educación para el Desarrollo y la sublínea CED-1.6: La gestión y calidad en educación, alineándose con la Estrategia Nacional de Educación "Bendiciones y Victorias 2024-2026". Se espera que esta herramienta tecnológica contribuya al bienestar físico, mental y académico de los estudiantes, mejorando los procesos de gestión institucional y la calidad educativa.

Palabras claves: hábitos saludables, registro escolar, sistema automatizado, Programación Orientada a Objetos, estadística descriptiva, gestión educativa

INTRODUCCIÓN

La promoción de hábitos saludables en el ámbito escolar es fundamental para el desarrollo integral de los estudiantes, ya que influye directamente en su bienestar físico, mental y rendimiento académico. En Nicaragua, la Estrategia Nacional de Educación "Bendiciones y Victorias 2024-2026" establece en su Eje 1 "Educación para la Vida" la necesidad de implementar programas integrales de hábitos saludables en las instituciones educativas. Sin embargo, muchas escuelas carecen de herramientas tecnológicas que faciliten el seguimiento sistemático de estos hábitos. El Instituto Nacional 21 de Junio promueve activamente hábitos de nutrición, ejercicio e higiene entre sus estudiantes, pero enfrenta limitaciones significativas debido a que los registros se realizan manualmente, sin una base de datos centralizada que permita consultar historiales, generar reportes automáticos o realizar análisis estadísticos. Ante esta problemática, el presente proyecto propone el diseño e implementación de un sistema automatizado de escritorio que permita registrar, monitorear y analizar el cumplimiento de hábitos saludables, contribuyendo así a la mejora de la gestión institucional y la calidad educativa.

ÍNDICE

TEMA **7**
1. OBJETIVOS **8**
1.1. Objetivo general 8
1.2. Objetivos específicos 8
2. DESCRIPCIÓN DE LA PROBLEMÁTICA **9**
3. PLANIFICACIÓN SCRUM **13**
3.1. Tabla de funcionalidades 13
3.2. Roles según Scrum 14
3.3. Historias de usuario 15
3.4. Product Backlog 40
3.5. Sprint Backlog 43
Sprint Backlog 1 43
Sprint Backlog 2 45
Sprint Backlog 3 46
Sprint Backlog 4 48
Sprint Backlog 5 50
Sprint Backlog 6 52
Sprint Backlog 7 54
Sprint Backlog 8 56
Sprint Backlog 9 58
Sprint Backlog 10 60
4. MAPEO DE PROCESOS DEL SISTEMA **63**
5. MODELO FÍSICO DE BD POBLADO CON INFORMACIÓN REAL (DIAGRAMA Y CAPTURAS) **63**
5.1. Diagrama ER 63
5.2. Script SQL creación de tablas 65
5.3. Datos de prueba 67
5.4. Tablas con datos reales 69
6. CONSULTAS DISEÑADAS A LA BASE DE DATOS **72**
7. DISEÑO DE ARQUITECTURA DE APLICACIÓN **74**
8. DIAGRAMAS UML **74**

8.1. Diagrama de clases 74
8.2. Diagrama de caso de uso 75
8.3. Diagrama de actividades 76
9. CODIGO DE CLASES DE ENTIDAD (CAPAS DE DATOS) 78
9.1. Clase Grado 78
9.2. Clase Sección 78
9.3. Clase Estudiante 79
9.4. Clase Hábito 81
9.5. Clase Usuario 82
9.6. Clase Registro 84
9.7. Clase Reporte 86
10. DISEÑO FUNCIONAL DEL PROTOTIPO (ENLACE FIGMA) 88
11. IMPLEMENTACIÓN DE OPERACIONES CRUD (CÓDIGO DOCUMENTADAS) 88
11.1. GradoDAO 88
11.2. SecciónDAO 89
11.3. EstudianteDAO 89
11.4. HábitoDAO 89
11.5. UsuarioDAO 90
11.6. RegistroDAO 90
11.7. ReporteDAO 90
12. REFERENCIAS BIBLIOGRÁFICAS 92
13. ANEXOS 94
13.1. Informe de sesiones de trabajos 94
13.2. Cronograma 96

TEMA

Desarrollo de un sistema de escritorio para el Instituto Nacional 21 de Junio.

1. OBJETIVOS

1.1. Objetivo general

Desarrollar un sistema de escritorio para el Instituto Nacional 21 de Junio que automatice el registro, monitoreo y análisis del seguimiento de hábitos saludables en los estudiantes durante el año lectivo 2026.

1.2. Objetivos específicos

- Identificar las necesidades y deficiencias actuales en el seguimiento de hábitos saludables mediante un análisis de los procesos y metodologías empleadas por el instituto.
- Diseñar un sistema de escritorio seguro que centralice la información sobre los hábitos saludables de los estudiantes
- Crear una interfaz de monitoreo en tiempo real que permita visualizar el progreso individual y grupal de los estudiantes en cuanto a hábitos saludables, registrando su constancia y su evolución académica y personal.

2. DESCRIPCIÓN DE LA PROBLEMÁTICA

Falta de seguimiento de hábitos saludables en estudiantes.

La falta de seguimiento y control de hábitos saludables en los estudiantes dificulta la identificación de comportamientos que afectan su bienestar. Por ello se propone automatizar este proceso mediante un sistema de escritorio que permita registrar, monitorear y evaluar hábitos como la alimentación, actividades físicas, y el descanso, facilitando la toma de decisiones dentro de la institución.

Según la Organización Mundial de la Salud (OMS, 2023), la promoción de la salud en el entorno escolar es una de las estrategias más efectivas para prevenir enfermedades crónicas no transmisibles, ya que la infancia y adolescencia son etapas clave para la adquisición de comportamientos que perdurarán en la vida adulta. En este sentido, Hernández-Beltrán et al. (2022) señalan que las intervenciones educativas sobre nutrición y actividad física realizadas en los centros educativos resultan positivas en la mejora de los hábitos de alimentación y la práctica deportiva entre los escolares, combinando ambas temáticas para prevenir la obesidad y el sedentarismo.

El Instituto Nacional 21 de Junio es una institución educativa de nivel secundario donde se presenta la falta de seguimiento de hábitos saludables en los estudiantes. Ante esta situación, se propone desarrollar el sistema para mejorar el registro, control y monitoreo de dichos hábitos.

Por otra parte, el área académica y de orientación, conformada por docentes y directora será la encargada de hacer uso del sistema para dar seguimiento a los hábitos saludables y apoyar al bienestar de los estudiantes.

Actualmente, el Instituto Nacional 21 de Junio promueve hábitos saludables entre sus estudiantes mediante charlas, actividades extracurriculares y recomendaciones generales, sin embargo, el seguimiento del cumplimiento de estos hábitos se realiza de manera manual, no existe una herramienta tecnológica que permita centralizar la información, generar reportes

de cumplimiento, ni analizar estadísticamente el impacto de los hábitos en el bienestar estudiantil, ya que los datos los datos no se encuentran sistematizados, lo que dificulta la evaluación y la identificación de estudiantes que requieren mayor acompañamiento.

El proceso de seguimiento de hábitos del instituto presenta múltiples problemas y necesidades que afectan su eficacia, entre los que destacan la falta de sistematización al no existir una herramienta tecnológica que centralice los registros de hábitos de todos los estudiantes, un seguimiento ineficiente debido a que los métodos manuales (hojas de papel, cuadernos) generan pérdida de información, duplicidad de registros y dificultad para consultar datos históricos, la ausencia de análisis estadístico que impide calcular promedios de cumplimiento o generar indicadores para evaluar el programa, la falta de reportes automatizados en formatos como PDF que faciliten la presentación de resultados a padres de familia o instancias superiores, la dificultad para la toma de decisiones institucionales al no contar con datos objetivos que permitan identificar qué áreas de intervención (nutrición, ejercicio o higiene) requieren mayor atención, la desvinculación entre el cumplimiento de hábitos y el rendimiento académico de los estudiantes al no poder relacionar ambas variables, y finalmente la ineficiencia en la asignación de recursos porque la institución no puede priorizar inversiones en las áreas de mayor necesidad ante la falta de datos concretos y confiables.

A nivel internacional, se han implementado diversos programas de intervención para promover hábitos saludables en el entorno escolar. Hernández-Beltrán et al. (2022) realizaron una revisión sistemática de programas de intervención en escolares españoles, encontrando que la mayoría de los programas analizados demuestran eficacia en la modificación de hábitos alimentarios y el aumento del nivel de actividad física. En el contexto latinoamericano, Viñamagua Chicaiza et al. (2026) abordaron los determinantes socioculturales de la salud en adolescentes, destacando la necesidad de estrategias educativas contextualizadas. En Nicaragua, la UNAN-Managua (2024) ha documentado proyectos pedagógicos como "Comer bien, es vivir mejor", demostrando el compromiso institucional con la promoción de hábitos saludables desde la formación docente.

El presente proyecto de investigación e innovación tecnológica se enmarca dentro de las líneas estratégicas establecidas por la UNAN-Managua en su agenda de investigación,

respondiendo a las necesidades del contexto educativo nacional y alineándose con las políticas públicas del país.

El Gobierno de Nicaragua, a través del Ministerio de Educación (MINED), ha establecido esta estrategia como marco rector de las políticas educativas del país. Contempla en su Eje 1: Educación para la Vida, específicamente en el Lineamiento 9, la Acción 1.1.19 que establece la implementación de un "Programa integral de hábitos saludables para el bienestar individual, familiar y comunitario".

El MINED también ha implementado el Programa Integral de Nutrición Escolar (PINE), que tiene como objetivo garantizar la alimentación saludable de los estudiantes en los centros educativos públicos del país, contribuyendo a mejorar su rendimiento académico y reducir la deserción escolar.

El Gobierno de Nicaragua, a través de los Ministerios de Salud (MINSA) y Educación (MINED), lanzó en 2025 la Estrategia Escuela Saludable, que promueve entornos educativos saludables mediante actividades de nutrición, higiene, actividad física y prevención de enfermedades.

Línea de Investigación UNAN-Managua:

CED-1: Educación para el Desarrollo

Esta línea de investigación aborda los procesos educativos desde una perspectiva integral, considerando la formación de los estudiantes no solo en aspectos académicos sino también en competencias para la vida, valores, salud y bienestar. El desarrollo de un sistema de registro de hábitos saludables se inscribe en esta línea porque busca fortalecer la educación integral de los estudiantes del Instituto Nacional 21 de Junio, promoviendo su bienestar físico, mental y académico a través del monitoreo sistemático de conductas saludables.

Sublínea de Investigación:

CED-1.6: La gestión y calidad en educación

Esta sublínea se enfoca en el estudio y mejora de los procesos de gestión institucional y la calidad educativa. El sistema propuesto contribuye a esta sublínea porque automatiza y optimiza los procesos de seguimiento de hábitos saludables permitiendo una gestión más eficiente de la información facilitando la toma de decisiones basada en datos objetivos y mejorando la calidad de calificaciones

3. PLANIFICACIÓN SCRUM

3.1. Tabla de funcionalidades

No	Tipo de usuario	Funcionalidad	Descripción	Encargado
1	Administrador	Gestión de estudiantes	Registrar, consultar, actualizar y dar de baja estudiantes	
2	Administrador	Gestión de hábitos	Crear, editar y desactivar hábitos del catálogo	
3	Administrador	Generación de reportes	Generar reportes PDF por periodo y para el hogar	
4	Administrador	Análisis estadístico	Calcular promedios y generar gráficos de cumplimiento	
5	Docente	Registro diario de hábitos	Registrar nutrición, ejercicio e higiene y otros de cada estudiante	
6	Docente	Consulta de cumplimiento	Consultar porcentajes de cumplimiento por estudiante o grupo	
	Administrador	Gestión de grados y secciones	Crear, modificar y eliminar grados y secciones	

Nota: Las funcionalidades fueron definidas con base en el análisis de la situación actual y los requerimientos identificados en el Instituto Nacional 21 de Junio. Los tipos de usuario corresponden a los roles establecidos en el sistema (Administrador y Docente).

3.2. Roles según Scrum

Rol	Nombres y Apellidos completos	Funciones
Scrum Master	Nerybeth Auxiliadora Bravo López	Facilita las reuniones de Scrum, elimina impedimentos y asegura que el equipo siga la metodología.
Product Owner	Yeison Alexander Gómez Morales	Define los requerimientos, prioriza el Product Backlog y valida las funcionalidades entregadas.
Development Team	Pablo Francisco Bravo Centeno	Desarrollador Dev Fullstack, Tester
	Anderson Johan Gonzalez Torrez	Desarrollador Dev Backend, Dev Frontend

Stakeholders	Agustina Rocha	Cliente final que recibe el producto

Nota: Roles asignados según metodología ágil Scrum para la gestión del desarrollo del proyecto.

3.3. Historias de usuario

HU1 - Registro de Consumo de Agua Diaria			
Como	Docente		
Quiero	Registrar la cantidad de vasos de agua (o mililitros) que consumo durante la jornada escolar		
Para	Mantener un seguimiento de hidratación diaria y asegurar que cumplen con los niveles recomendados para el bienestar físico y mental		
Validación	• Garantizar que el sistema ingrese valores numéricos (vasos) • Mostrar en el sistema el total acumulado del día en tiempo real. • Validar el total de minutos activos de la semana	**Impacto**	2
		Prioridad	1
		Estimación	1 hrs

<table>
<tr><th colspan="4">HU2 - Registro de actividad física</th></tr>
<tr><td>Como</td><td colspan="3">Docente</td></tr>
<tr><td>Quiero</td><td colspan="3">Registrar el tipo de ejercicio y el tiempo en minutos que dedican a moverse cada día</td></tr>
<tr><td>Para</td><td colspan="3">Monitorear el nivel de sedentarismo y motivar a cumplir con las recomendaciones de salud física escolar</td></tr>
<tr><td rowspan="3">Validación</td><td rowspan="3">•¨verificar que el sistema seleccione el tipo de actividad (correr, caminar, deporte, etc.) de una lista predefinida.

• verificar que el sistema ingrese la duración en minutos (solo números enteros).

• Calcular el total de minutos activos de la semana</td><td>Impacto</td><td>2</td></tr>
<tr><td>Prioridad</td><td>1</td></tr>
<tr><td>Estimación</td><td>1 hrs</td></tr>
</table>

HU3 - Registro de Horas de Sueño			
Como	Docente		
Quiero	Registrar la hora de dormír y la hora de despertar cada día		
Para	Evaluar si están descansando lo suficiente para rendir académicamente y detectar patrones de cansancio		
Validación	• Calcular automáticamente el total de horas dormidas. • Mostrar una alerta si las horas de sueño son menores a 7 u 8 horas (lo recomendado para estudiantes). • Permitir que el sistema ingrese valores numéricos (horas)	**Impacto**	2
		Prioridad	1
		Estimación	2 hrs

HU4-Registrar hábitos

Como	Administrador		
Quiero	registrar, editar y eliminar los hábitos a monitorear		
Para	Definir y mantener actualizado los aspectos de salud que serán evaluados en los estudiantes		
Validación	• Mostrar formulario con campos obligatorios. • Validar campos y guardar correctamente el hábito en la base de datos • Mostrar mensaje de confirmación de cambios	**Impacto**	32
		Prioridad	5
		Estimación	1 hrs

HU5-Registrar estudiantes	
Como	Docente
Quiero	Registrar los datos de los estudiantes
Para	Poder registrarlos en el sistema y realizar el seguimiento de sus hábitos

Validación	• Mostrar formulario con campos obligatorios.	**Impacto**	32
	• Validar campos y guardar correctamente en la base de datos	**Prioridad**	5
	• Mostrar mensaje de confirmación del registro	**Estimación**	2hrs

HU6- Registrar aulas			
Como	Administrador		
Quiero	Registrar y editar las aulas disponibles en el instituto		
Para	Asignar los estudiantes a sus respectivos grupos y poder filtrar el monitoreo de hábitos por aula		
Validación	• Mostrar formulario con campos obligatorios.	**Impacto**	32
	• Validar campos y guardar correctamente en la base de datos	**Prioridad**	5
	• Mostrar mensaje de confirmación del registro o cambios	**Estimación**	1 hrs

HU7- Ingresar cumplimientos			
Como	Docente		
Quiero	Ingresar el cumplimiento de hábitos de mis estudiantes		
Para	Llevar el registro sistemático		
Validación	• Mostrar secciones disponibles • Mostrar formulario con campos obligatorios. • Validar campos y guardar correctamente en la base de datos • Mostrar mensaje de confirmación del registro	**Impacto**	32
		Prioridad	4
		Estimación	1hr

HU8 - Metas Personalizadas	
Como	Docente
Quiero	Establecer objetivos semanales personalizados
Para	Crear una rutina constante y desarrollar disciplina de manera progresiva.

Validación	• Generar formulario con campos obligatorios • Analizar metas predefinidas o escribir una personalizada y guardarla. • Mostrar mensaje de confirmación	**Impacto**	2
		Prioridad	3
		Estimación	2hrs

HU9- Recordatorio	
Como	Docente
Quiero	Recibir notificaciones sobre el registro de hábitos

Para	No olvidar registrar los cumplimientos de los estudiantes.		
Validación	• Mostrar formulario con campos obligatorios • Validar la fecha programada • Mostrar mensaje de confirmación	**Impacto**	32
		Prioridad	3
		Estimación	1hr

HU10- Logros y Medallas			
Como	Docente		
Quiero	Configurar y asignar logros o medallas a los estudiantes según su cumplimiento de hábitos		
Para	Motivar a los estudiantes mediante un sistema de recompensas que reconozca su esfuerzo y constancia		
Validación	• Confirmar que el sistema calcule correctamente el cumplimiento de requisitos para asignar logros automáticamente. • Validar que no existan logros duplicados con el mismo nombre.	**Impacto**	4
		Prioridad	2

	• Validar que las imágenes o íconos de medallas tengan un formato válido	**Estimación**	2hrs

HU11- Tips de Salud			
Como	Estudiante		
Quiero	Visualizar tips de salud dentro del sistema		
Para	Mejorar los hábitos y bienestar diario		
Validación	• Mostrar tips de salud de forma clara • Generar recomendaciones sobre bienestar físico y emocional • Determinar los tips periódicamente • Permitir visualizar los tips fácilmente	**Impacto**	4
		Prioridad	2
		Estimación	

<table>
<tr><th colspan="4">HU12 - Privacidad de datos</th></tr>
<tr><td>Como</td><td colspan="3">Docente</td></tr>
<tr><td>Quiero</td><td colspan="3">Que los registros emocionales sean privados y estén protegidos</td></tr>
<tr><td>Para</td><td colspan="3">Garantizar la confidencialidad de los datos</td></tr>
<tr><td rowspan="3">Validación</td><td rowspan="3"><ul><li>Asegurar acceso según el tipo de usuario</li><li>Asegurar la información personal del estudiante</li><li>Asegurar acceso solo a usuarios autorizados</li><li>Garantizar seguridad en el sistema</li></ul></td><td>Impacto</td><td>32</td></tr>
<tr><td>Prioridad</td><td>3</td></tr>
<tr><td>Estimación</td><td>3hrs</td></tr>
</table>

<table>
<tr><th colspan="4">HU13 - Perfil personal</th></tr>
<tr><td>Como</td><td colspan="3">Administrador</td></tr>
<tr><td>Quiero</td><td colspan="3">Tener un perfil personal en el sistema de cada estudiante</td></tr>
<tr><td>Para</td><td colspan="3">Visualizar la información sobre el bienestar</td></tr>
<tr><td rowspan="3">Validación</td><td rowspan="3"><ul><li>Mostrar información personal del estudiante</li><li>Visualizar indicadores de bienestar</li><li>Confirmar actualización de datos básicos</li></ul></td><td>Impacto</td><td>32</td></tr>
<tr><td>Prioridad</td><td>3</td></tr>
<tr><td>Estimación</td><td>3hrs</td></tr>
</table>

HU14- Gestion de usuarios			
Como	Administrador		
Quiero	Gestionar los usuarios del sistema		
Para	Controlar accesos y roles		
Validación	• Confirmar que el sistema cree, editar roles y eliminar usuarios. • Asignar roles (docente, estudiantes, padres de familia). • Determinar acceso según rol.	**Impacto**	32
		Prioridad	3
		Estimación	2hrs

HU15- Accesos para padres

Como	Administrador		
Quiero	Un espacio para padres en el sistema		
Para	Consultar el bienestar de el hijo		
Validación	• Permitir inicio de sesión seguro para padres • Mostrar información del bienestar del estudiante • Autorizar sólo la información de su hijo • Visualizar indicadores de bienestar	**Impacto**	2
		Prioridad	2
		Estimación	5hrs

HU16- Filtros de búsquedas	
Como	Docente
Quiero	Aplicar filtros de búsquedas sobre el bienestar de los estudiantes
Para	Encontrar información específica de forma rápida

Validación	• Permitir búsqueda por nombre del estudiante • Filtrar por nivel de bienestar • Filtrar por estado general • Mostrar resultados automáticamente • Permitir combinar varios filtros	**Impacto**	32
		Prioridad	3
		Estimación	2hrs

HU17- Dashboard comparativo			
Como	Docente		
Quiero	Visualizar el bienestar del estudiante		
Para	Analizar y comparar su estado de bienestar		
Validación	• Mostrar gráficos comparativos del bienestar • Permitir seleccionar uno o varios estudiantes • Visualizar indicadores de bienestar	**Impacto**	32

	• Actualizar los datos automáticamente		
		Prioridad	2
		Estimación	8hrs

HU18- Reportes para el hogar			
Como	Padre de familia		
Quiero	Obtener reportes sobre los hábitos saludables del estudiante		
Para	Dar seguimiento desde casa		
Validación	• Generar reportes claros sobre hábitos saludables. • Permitir descargar los reportes. • Mostrar información de su bienestar saludable.	**Impacto**	4

	• Autenticar que el sistema haga las observaciones o recomendaciones. • Autorizar acceso solo para padres autorizados.		
		Prioridad	2
		Estimación	6hrs

HU19- Anuncios de bienestar			
Como	Administrador		
Quiero	Publicar anuncios de bienestar		
Para	Promover la salud en los estudiantes		
Validación	• Cargar, editar y eliminar anuncios	**Impacto**	3

	• Mostrar los anuncios a los estudiantes. • Notificarme cuando haya anuncios nuevos.		
		Prioridad	3
		Estimación	4hrs

HU20- Inicio de sesión	
Como	Administrador
Quiero	Iniciar sesión con credenciales seguras
Para	Acceder a las funcionalidades según el rol y garantizar que solo personal autorizado pueda usar el sistema

Validación	• Mostrar formulario con campos obligatorios • Mostrar diferentes menús y funcionalidades según el rol • Volver a la pantalla login al cerrar sesión	**Impacto**	4
		Prioridad	3
		Estimación	2hrs

HU 21- Panel general			
Como	Administrador		
Quiero	quiero ver un resumen grupal del cumplimiento de hábitos		
Para	identificar tendencias en el salón		
Validación	• Generar el resumen • Procesar las tendencias • Actualizar los datos	**Impacto**	4
		Prioridad	3
		Estimación	4hrs

HU22- Alertas de riesgo			
Como	Administrador		
Quiero	quiero que el sistema me alerte si un estudiante registra falta de sueño o desánimo constante		
Para	para intervenir a tiempo.		
Validación	• Identificar cuando un estudiante registre falta de sueño o desánimo • Generar las alertas • Mostrar mensaje de alerta	**Impacto**	32
		Prioridad	2
		Estimación	5hrs

HU23- generación de reportes	
Como	Administrador

Quiero	quiero exportar reportes mensuales de salud estudiantil		
Para	para presentarlos a la dirección.		
Validación	• Generación de reportes • Exportar en PDF	**Impacto**	32
		Prioridad	2
		Estimación	2hrs

HU24- gestión de alumnos			
Como	Administrador		
Quiero	quiero poder dar de alta o baja a los estudiantes		
Para	para mi sección dentro del sistema.		
Validación	• Registrar los estudiantes de (Alta) • Eliminar estudiantes de (Baja)	**Impacto**	4
	• Mostrar lista de alumnos	**Prioridad**	3

		Estimación	4hrs

HU25- comparativa académica			
Como	Administrador		
Quiero	quiero cruzar datos de hábitos con promedios de notas		
Para	para analizar la correlación entre salud y éxito académico.		
Validación	• Cargar de datos académicos • Calcular promedio de notar y hábitos • Visualizar comparativa	**Impacto**	16
		Prioridad	5
		Estimación	6hrs

HU26- configuración de hábitos	
Como	Administrador
Quiero	quiero definir qué hábitos son obligatorios de registrar

Para	para todos los niveles escolares.		
Validación	• Determinar de hábitos obligatorios • Analizar por nivel escolar • Determinar la que sea obligatorio	**Impacto**	2
		Prioridad	3
		Estimación	3hrs

HU27 - Bitácora de Alimentación Saludable	
Como	Docente
Quiero	Registrar el consumo de grupos de alimentos saludables (frutas, verduras, proteínas) y si han evitado la comida ultra procesada durante el día
Para	Ser consciente de la calidad nutricional y mejorar decisiones alimenticias dentro y fuera del instituto

Validación	• Generar una lista de verificación por categoría. • Añadir una nota breve sobre el almuerzo, merienda escolar. • Mostrar campos obligatorios	**Impacto**	2
		Prioridad	2
		Estimación	3hrs

HU28 - Registro de Estado de Ánimo Diario			
Como	Docente		
Quiero	Seleccionar un indicador visual, que represente el sentir emocional cada día.		
Para	Ser consciente de la salud mental y permitir que el sistema detecte patrones de desánimo que puedan afectar el rendimiento		
Validación	• Mostrar una interfaz con al menos 5 opciones visuales (ej. Muy feliz,	**Impacto**	2

HU28 - Registro de Estado de Ánimo Diario			
	Feliz, Neutral, Triste, Estresado). ● Añadir un comentario breve opcional sobre el motivo de su estado de ánimo. ● Guardar la fecha y hora exacta del registro.		
		Prioridad	2
		Estimación	2hrs

HU29 - Visualización de Progreso y Estadísticas	
Como	Docente
Quiero	Ver gráficas y resúmenes de los hábitos registrados (agua, sueño, ejercicio, etc.) de la última semana o mes.
Para	Analizar comportamiento, identificar en qué áreas mejorar

Validación	• Mostrar al menos un gráfico de barras o líneas por cada hábito. • Filtrar la vista por "Esta semana" o "Este mes". • Asegurar que las gráficas deben ser fáciles de leer, usando colores diferenciadores para cada categoría.	**Impacto**	16
		Prioridad	5
		Estimación	6hrs

HU30-Auditoría de actividades			
Como	Administrador		
Quiero	consultar un registro de auditoría con todas las acciones realizadas en el sistema		
Para	supervisar el uso del sistema y detectar acciones no autorizadas		
Validación	• Registrar que cada acción quede con usuario, fecha, hora y acción realizada. • Comprobar que el registro de auditoría no pueda ser	**Impacto**	16

	modificado ni eliminado por usuarios regulares. ● Filtrar consulta por usuario, fecha y tipo de acción		
		Prioridad	5
		Estimación	5hrs

3.4. Product Backlog

Product Backlog

ID	**Historia de usuario**	**Importancia**	**Sprint Programado**	**Fecha de Inicio**	**Fecha Fin**
HU1	Registro de Consumo de Agua Diaria	2	10		
HU2	Registro de actividad física	2	10		
HU3	Registro de Horas de Sueño	2	10		
HU4	Registrar hábitos	160	1		
HU5	Registrar estudiantes	160	1		
HU6	Registrar aulas	160	1		
HU7	Ingresar cumplimientos	128	2		
HU8	Metas Personalizada	6	8		

ID	Historia de usuario	Importancia	Sprint Programado	Fecha de Inicio	Fecha Fin
	s				
HU9	Recordatorio	96	3		
HU10	Logros y Medallas	8	7		
HU11	Tips de Salud	8	7		
HU12	Privacidad de datos	128	2		
HU13	Perfil personal	96	3		
HU14	Gestion de usuarios	128	2		
HU15	Accesos para padres	4	9		
HU16	Filtros de búsquedas	96	3		
HU17	Dashboard comparativo	64	5		
HU18	Reportes para el hogar	8	7		
HU19	Anuncios de bienestar	6	8		
HU20	Inicio de sesión	12	6		
HU21	Panel general	12	6		
HU22	Alertas de riesgo	64	5		
HU23	generación de reportes	64	5		

ID	Historia de usuario	Importancia	Sprint Programado	Fecha de Inicio	Fecha Fin
HU24	Gestión de alumnos	12	6		
HU25	comparativa académica	80	4		
HU26	configuración de hábitos	6	8		
HU27	Bitácora de Alimentación Saludable	4	9		
HU28	Registro de Estado de Ánimo Diario	4	9		
HU29	Visualización de Progreso y Estadísticas	80	4		
HU30	Auditoría de actividades	80	4		

3.5. Sprint Backlog

Sprint Backlog 1

ID	Historia de Usuario	Tareas	Tiempo	Responsable
HU4	Registrar hábitos	Diseñar interfaz	1hr	
		Programar formulario con campos obligatorios	4hr	
		Implementar funciones de guardar, editar, eliminar y actualizar	2hr	
		Programar base de datos	4hr	
		Pruebas de funcionalidad	1hr	

HU5	Registrar estudiantes	Diseñar interfaz	1hr	
		Programar formulario con campos obligatorios	4hr	
		Implementar funciones de guardar	2hr	
		Programar base de datos	4hr	
		Pruebas de funcionalidad	1hr	
HU6	Registrar aulas	Diseñar interfaz	1hr	
		Programar formulario con campos obligatorios	4hr	
		Implementar funciones de guardar, editar y actualizar	2hr	

		Programar base de datos	4hr	
		Pruebas de funcionalidad	1hr	

Sprint Backlog 2

ID	Historia de Usuario	Tareas	Tiempo	Responsable
HU7	Ingresar cumplimientos	Diseñar interfaz	1hr	
		Implementar validaciones de campos obligatorios	2hr	
		Programar guardado en base de datos	4hr	
		Realizar pruebas de seguridad	1hr	

HU12	Privacidad de datos	Diseñar políticas de privacidad del sistema	1hr	
		Programar para proteger los datos en la base de datos	4hr	
		Realizar pruebas de seguridad	1hr	
HU14	Gestion de usuarios	Diseñar interfaz para asignar rol a usuario	1hr	
		Programar autenticación contra base de datos	4hrs	
		Programar validación de permisos en cada módulo	4hrs	
		Pruebas de funcionalidad	1hr	

Sprint Backlog 3

ID	**Historia de Usuario**	**Tareas**	**Tiempo**	**Responsable**

HU9	Recordatorio	Diseño de la interfaz	1hr	
		Programar validación de campos obligatorios	2hr	
		Programar validación de fecha y hora programada	2hr	
		Programar mensaje de confirmación: "Recordatorio automático configurado"	2hr	
		Pruebas	1hr	
HU13	Perfil personal	Diseño de la interfaz	1hr	
		Programar edición y actualización de datos del perfil	2hr	
		Programar base de datos	2hr	

		Pruebas	1hr	
H16	Filtros de búsquedas	Diseño de la interfaz	1hr	
		Programar búsqueda por nombre	2hr	
		Programar filtro por nivel de bienestar y por estado general	2hr	
		Pruebas	1hr	

Sprint Backlog 4

ID	**Historia de Usuario**	**Tareas**	**Tiempo**	**Responsable**
HU25	comparativa académica	Diseño de la interfaz	1hr	
		Programar	2hr	

		comparativa de evolución entre hábitos y calificaciones por parcial		
		Programar visualización en gráfico de barras comparativas	2hr	
		Programar exportación de comparativa (PDF o imagen)	2hr	
		Pruebas	1hr	
HU29	Visualización de Progreso y Estadísticas	Diseño de la interfaz	1hr	
		Programar cálculo de porcentaje de cumplimiento por estudiante	2hr	
		Calcular promedio general del grupo	2hr	
		Programar estadísticas por estudiante y grado	2hr	
		Pruebas	1hr	

HU30	Auditoría de actividades	Diseño de la interfaz	1hr	
		Programar registro automático de acciones	2hr	
		Programar registro de inicio y cierre de sesión	2hr	
		Programar filtros de auditoría	2hr	
		Programar visualización de historial	2hr	
		Programar imposibilidad de modificar y eliminar las acciones de auditoría	2hr	
		Pruebas	1hr	

Sprint Backlog 5

ID	Historia de Usuario	Tareas	Tiempo	Responsable
HU17	Dashboard comparativo	Diseño de la interfaz	1hr	
		Programar selector múltiple de estudiantes para comparar	2hr	
		Programar comparativa de hábitos más cumplidos y menos cumplidos	2hr	
		Actualización en tiempo real de datos del dashboard	2hr	
		Pruebas	1hr	
HU22	Alertas de riesgo	Diseño de la interfaz	1hr	
		Implementar	2hr	

		detección automática de estudiantes con bajo cumplimiento		
		Programar envío de alertas	2hr	
		Programar historial de alertas enviadas	2hr	
		Pruebas	1hr	
HU23	generación de reportes	Diseño de la interfaz	1hr	
		Programar validación de parámetros	2hr	
		Programar reporte de cumplimiento individual y grupal por semana	2hr	
		Programar exportación a PDF y Excel	2hr	
		Pruebas	1hr	

Sprint Backlog 6

ID	Historia de Usuario	Tareas	Tiempo	Responsable
HU20	Inicio de sesión	Diseño de la interfaz	1hr	
		Programar validación de campos obligatorios	2hr	
		Programar redirección según rol	2hr	
		Pruebas	1hr	
HU21	Panel general	Diseño de la interfaz		
		Programar visualización de resumen diario	2hr	
		Visualización de notificaciones recientes	2hr	
		Programar accesos directos a funciones principales	2hr	

		Programar calendario	2hr	
		Pruebas	1hr	
HU24	Gestión de alumnos	Diseño de la interfaz	1hr	
		Programar validación de campos del formulario	2hr	
		Programar registro, eliminación y edición de alumno	2hr	
		Programar búsqueda y filtrado de alumnos por nombre o código	2hr	
		Pruebas	1hr	

Sprint Backlog 7

ID	**Historia de Usuario**	**Tareas**	**Tiempo**	**Responsable**
HU10	Logros y Medallas	Diseño de la interfaz	1hr	

		Programar validación de condiciones para desbloquear medallas	2hr	
		Implementar detección automática de cumplimiento de logros	2hr	
		Programar notificación al obtener nueva medalla	2hr	
		Programar galería de medallas por alcanzar	2hr	
		Pruebas	1hr	
HU11	Tips de Salud	Diseño de la interfaz	1hr	
		Programar programación automática de tips rotativos	2hr	
		Programar visualización de tips en panel general	2hr	
		Pruebas	1hr	
HU18	Reportes para el	Diseño de la interfaz	1hr	

	hogar	Programar reporte semanal de hábitos del estudiante	2hr	
		Añadir reporte de medallas obtenidas en el período	2hr	
		Programar envío automático por correo a los padres	2hr	
		Programar generación de reporte en PDF amigable	2hr	
		Pruebas	1hr	

Sprint Backlog 8

ID	**Historia de Usuario**	**Tareas**	**Tiempo**	**Responsable**
HU8	Metas Personalizadas	Diseño de la interfaz	1hr	
		Programar validación	2hr	

		de metas		
		Programar asignación de metas por estudiante o por grupo	2hr	
		Implementar seguimiento automático de progreso hacia la meta	2hr	
		Programar visualización de progreso	2hr	
		Programar notificación al alcanzar una meta	2hr	
		Programar historial de metas cumplidas	2hr	
		Pruebas	1hr	
HU19	Anuncios de bienestar	Diseño de la interfaz	1hr	
		Programar visualización de anuncios activos en panel general	2hr	
		Implementar programación de	2hr	

		publicación		
		Programar panel de administración de anuncios	2hr	
		Pruebas	1hr	
HU26	configuración de hábitos	Diseño de la interfaz	1hr	
		Programar asignación de hábitos obligatorios	2hr	
		Validar campos	2hr	
		Programar definición de frecuencia del hábito	2hr	
		Pruebas	2hr	

Sprint Backlog 9

ID	**Historia de Usuario**	**Tareas**	**Tiempo**	**Responsable**
HU15	Accesos para padres	Diseño de la interfaz	1hr	

		Programar inicio de sesión de visitas	2hr	
		Programar búsqueda por nombre o código	2hr	
		Programar visualización de progreso del estudiante	2hr	
		Pruebas		
HU27	Bitácora de Alimentación Saludable	Diseño de la interfaz	1hr	
		validación de campos	2hr	
		Programar selección de alimentos desde catálogo predefinido	2hr	
		Programar base de datos	2hr	
		Programar registro automático de fecha y hora		
		Pruebas	1hr	
HU28	Registro de Estado de Ánimo Diario	Diseño de la interfaz	1hr	
		Programar validación de selección de	2hr	

		ánimo		
		Programar campo opcional de nota o comentario	2hr	
		Programar opciones de ánimo	2hr	
		Programar base de datos	2hr	
		Programar registro automático de fecha y hora	2hr	
		Pruebas	1hr	

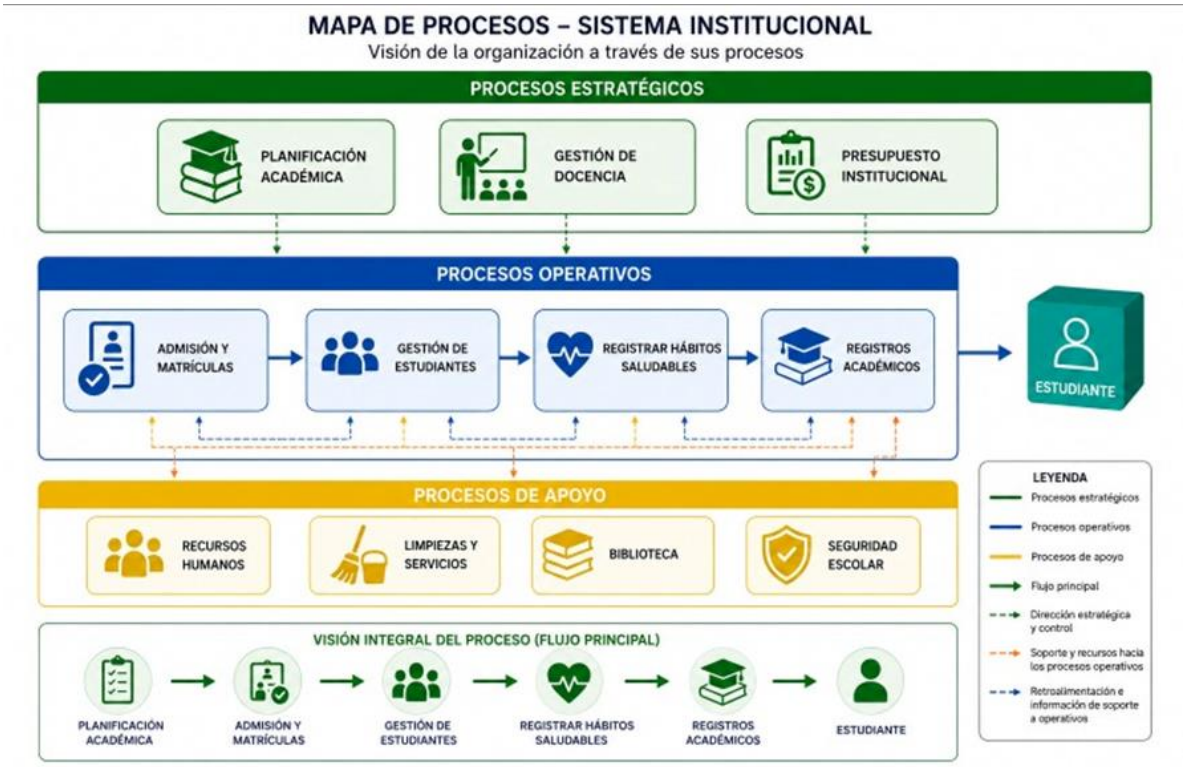
Sprint Backlog 10

ID	**Historia de Usuario**	**Tareas**	**Tiempo**	**Responsable**
HU1	Registro de Consumo de Agua Diaria	Diseño de la interfaz	1hr	
		Programar validación de cantidad	2hr	
		Programar meta	2hr	

		diaria personalizable		
		Implementar visualización de progreso diario	2hr	
		Programar registro automático de fecha y hora	2hr	
		Programar base de datos	2hr	
		Pruebas	1hr	
HU2	Registro de actividad física	Diseño de la interfaz		
		Programar validación de campos		
		Programar visualización de resumen semanal y racha de actividad		
		Programar registro automático de fecha y hora		
		Programar base de datos		
		Pruebas		
HU3	Registro de Horas de	Diseño de la interfaz		

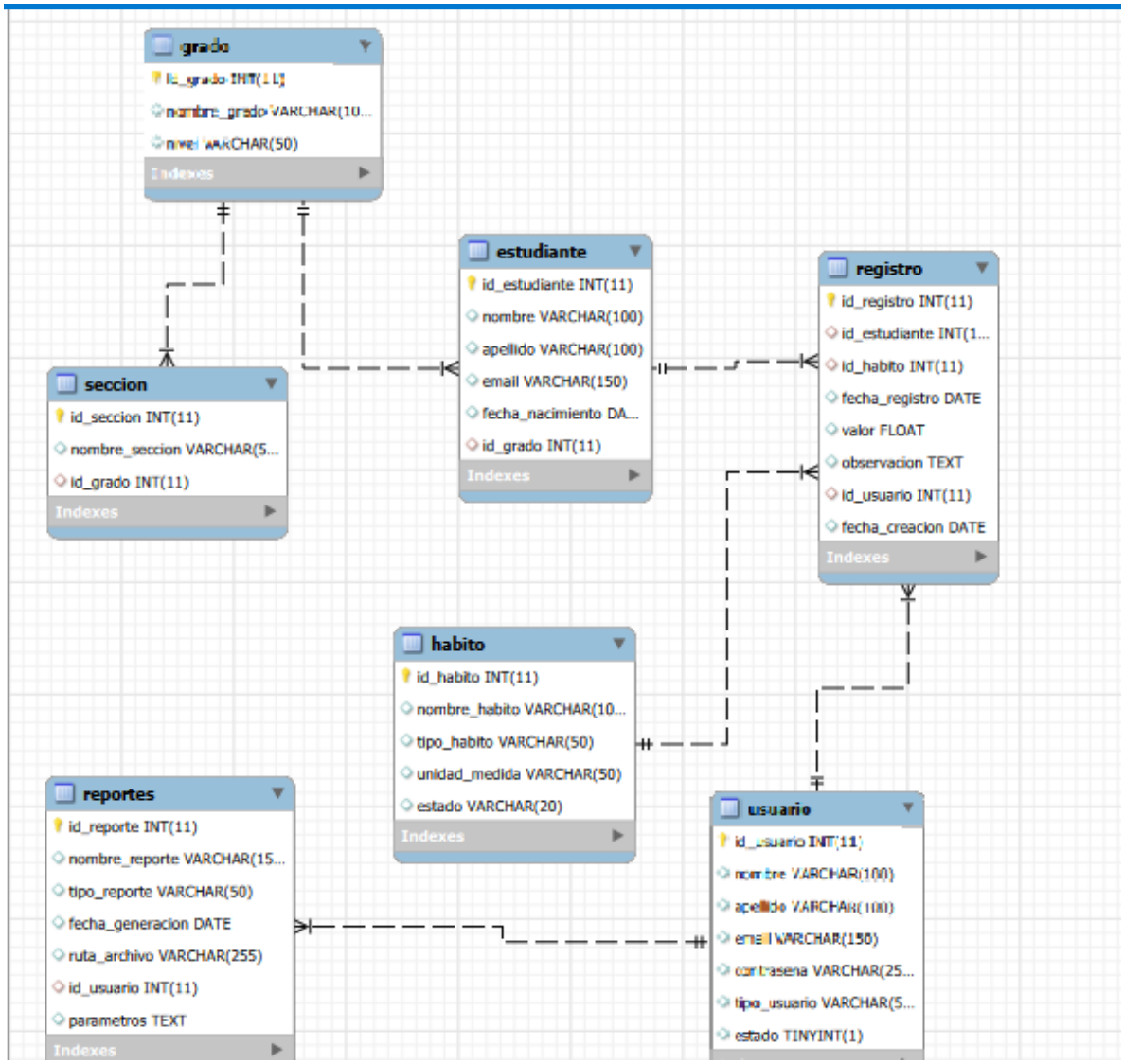
	Sueño	Programar validación de campos		
		Programar opción de calidad de sueño		
		Programar visualización de promedio semanal de sueño		
		Programar registro automático de fecha y hora		
		Programar base de datos		
		Pruebas		

4. MAPEO DE PROCESOS DEL SISTEMA



5. MODELO FÍSICO DE BD POBLADO CON INFORMACIÓN REAL (DIAGRAMA Y CAPTURAS)

5.1. Diagrama ER



5.2. Script SQL creación de tablas

```sql

CREATE DATABASE SistemaRegistroHabitos;
USE SistemaRegistroHabitos;

-- Tabla Grado
CREATE TABLE Grado (
    id_grado INT PRIMARY KEY AUTO_INCREMENT,
    nombre_grado VARCHAR(100),
    nivel VARCHAR(50)
);

-- Tabla Usuario
CREATE TABLE Usuario (
    id_usuario INT PRIMARY KEY AUTO_INCREMENT,
    nombre VARCHAR(100),
    apellido VARCHAR(100),
    email VARCHAR(150) UNIQUE,
    contrasena VARCHAR(255),
    tipo_usuario VARCHAR(50),
    estado BOOLEAN
);
```

```sql
22
23      --  Tabla Habito
24  CREATE TABLE Habito (
25      id_habito INT PRIMARY KEY AUTO_INCREMENT,
26      nombre_habito VARCHAR(100),
27      tipo_habito VARCHAR(50),
28      unidad_medida VARCHAR(50),
29      estado VARCHAR(20)
30  );
31
32      --  Tabla Seccion
33  CREATE TABLE Seccion (
34      id_seccion INT PRIMARY KEY AUTO_INCREMENT,
35      nombre_seccion VARCHAR(50),
36      id_grado INT,
37      FOREIGN KEY (id_grado) REFERENCES Grado(id_grado)
38  );
39
40      --  Tabla Estudiante
41  CREATE TABLE Estudiante (
42      id_estudiante INT PRIMARY KEY AUTO_INCREMENT,
43      nombre VARCHAR(100),
44      apellido VARCHAR(100),
45      email VARCHAR(150),
46      fecha_nacimiento DATE,
47      id_grado INT,
48      FOREIGN KEY (id_grado) REFERENCES Grado(id_grado)
49  );
50
51  -- Tabla Registro
52  CREATE TABLE Registro (
53      id_registro INT PRIMARY KEY AUTO_INCREMENT,
54      id_estudiante INT,
55      id_habito INT,
56      fecha_registro DATE,
57      valor FLOAT,
58      observacion TEXT,
59      id_usuario INT,
60      fecha_creacion DATE,
61      FOREIGN KEY (id_estudiante) REFERENCES Estudiante(id_estudiante),
62      FOREIGN KEY (id_habito) REFERENCES Habito(id_habito),
63      FOREIGN KEY (id_usuario) REFERENCES Usuario(id_usuario)
64  );
```

```sql
);
-- Tabla Reportes
CREATE TABLE Reportes (
    id_reporte INT PRIMARY KEY AUTO_INCREMENT,
    nombre_reporte VARCHAR(150),
    tipo_reporte VARCHAR(50),
    fecha_generacion DATE,
    ruta_archivo VARCHAR(255),
    id_usuario INT,
    parametros TEXT,
    FOREIGN KEY (id_usuario) REFERENCES Usuario(id_usuario)
);
```

5.3. Datos de prueba

```sql
INSERT INTO Grado (nombre_grado, nivel) VALUES
('Séptimo Grado', 'Secundaria'),
('Octavo Grado', 'Secundaria'),
('Noveno Grado', 'Secundaria'),
('Décimo Grado', 'Secundaria'),
('Undécimo Grado', 'Secundaria'),
('Séptimo Grado B', 'Secundaria'),
('Octavo Grado B', 'Secundaria'),
('Noveno Grado B', 'Secundaria'),
('Décimo Grado B', 'Secundaria'),
('Undécimo Grado B', 'Secundaria');

INSERT INTO Usuario (nombre, apellido, email, contrasena, tipo_usuario, estado) VALUES
('Ana', 'Martinez', 'ana.mtz@colegio.com', 'pass123', 'Administrador', 1),
('Carlos', 'Gomez', 'carlos.g@colegio.com', 'secure456', 'Docente', 1),
('Lucia', 'Lopez', 'lucia.l@colegio.com', 'admin789', 'Administrador', 1),
('Jorge', 'Rodriguez', 'jorge.r@colegio.com', 'clavedocente', 'Docente', 1),
('Elena', 'Perez', 'elena.p@colegio.com', 'perez2026', 'Docente', 1),
('David', 'Sánchez', 'david.s@colegio.com', 'david99', 'Docente', 1),
('Sofia', 'Diaz', 'sofia.d@colegio.com', 'sofia123', 'Administrador', 1),
('Miguel', 'Torres', 'miguel.t@colegio.com', 'torres456', 'Docente', 1),
('Laura', 'Castro', 'laura.c@colegio.com', 'laura789', 'Administrador', 1),
```

```
100     ('Pedro', 'Morales', 'pedro.m@colegio.com', 'pedro2026', 'Docente', 1);
101
102 •   INSERT INTO Habito (nombre_habito, tipo_habito, unidad_medida, estado) VALUES
103     ('Entrega de Ensayos', 'Académico', 'Porcentaje', 'Activo'),
104     ('Puntualidad en Bloque 1', 'Conducta', 'Minutos tarde', 'Activo'),
105     ('Participación en Debates', 'Académico', 'Interacciones', 'Activo'),
106     ('Uso Correcto de Celular', 'Disciplina', 'Llamadas de atención', 'Activo'),
107     ('Trabajo de Laboratorio', 'Académico', 'Calificación', 'Activo'),
108     ('Asistencia a Tutorías', 'Apoyo', 'Horas asistidas', 'Activo'),
109     ('Respeto en Comunidad', 'Convivencia', 'Incidentes', 'Activo'),
110     ('Cuidado del Mobiliario', 'Disciplina', 'Reportes de daño', 'Activo'),
111     ('Liderazgo en Proyectos', 'Social', 'Puntos acumulados', 'Activo'),
112     ('Uso de Uniforme Completo', 'Disciplina', 'Cumplimiento', 'Activo');
113
114 •   INSERT INTO Seccion (nombre_seccion, id_grado) VALUES
115     ('Única', 1),
116     ('Sección A', 2),
117     ('Sección B', 2),
118     ('Única', 3),
119     ('Sección A', 4),
120     ('Sección B', 4),
```

```
121     ('Matutina', 5),
122     ('Vespertina', 5),
123     ('Sección A', 6),
124     ('Única', 7);
125
126
127 •   INSERT INTO Estudiante (nombre, apellido, email, fecha_nacimiento, id_grado) VALUES
128     ('Juan', 'Alvarez', 'juan.alvarez@secundaria.com', '2014-04-12', 1),
129     ('Maria', 'Benitez', 'maria.benitez@secundaria.com', '2013-08-22', 2),
130     ('Luis', 'Cabrera', 'luis.cabrera@secundaria.com', '2012-01-15', 3),
131     ('Andres', 'Delgado', 'andres.delgado@secundaria.com', '2011-11-05', 4),
132     ('Rosa', 'Espinoza', 'rosa.espinoza@secundaria.com', '2010-06-30', 5),
133     ('Kevin', 'Flores', 'kevin.flores@secundaria.com', '2013-02-14', 6),
134     ('Camila', 'Gutierrez', 'camila.gutierrez@secundaria.com', '2012-09-18', 7),
135     ('Diego', 'Herrera', 'diego.herrera@secundaria.com', '2011-05-25', 8),
136     ('Natalia', 'Ibarra', 'natalia.ibarra@secundaria.com', '2010-12-01', 9),
137     ('Gabriel', 'Juarez', 'gabriel.juarez@secundaria.com', '2009-07-07', 10);
138
139 •   INSERT INTO Registro (id_estudiante, id_habito, fecha_registro, valor, observacion, id_usuario, fecha_creacion) VALUES
140     (1, 1, '2026-05-01', 0.0, 'Ausente', 2, '2026-05-01'),
141     (2, 4, '2026-05-01', 0.0, 'No cumplió en nada', 6, '2026-05-01'),
```

```
142     (3, 2, '2026-05-02', 15.0, 'Bajo rendimiento', 4, '2026-05-02'),
143     (4, 3, '2026-05-02', 0.0, 'Ausente', 4, '2026-05-02'),
144     (5, 5, '2026-05-03', 0.0, 'No cumplió en nada', 2, '2026-05-03'),
145     (6, 1, '2026-05-03', 4.0, 'Bajo rendimiento', 5, '2026-05-03'),
146     (7, 10, '2026-05-04', 0.0, 'Ausente', 6, '2026-05-04'),
147     (8, 7, '2026-05-04', 3.0, 'No cumplió en nada', 10, '2026-05-04'),
148     (9, 2, '2026-05-05', 25.0, 'Bajo rendimiento', 4, '2026-05-05'),
149     (10, 4, '2026-05-05', 0.0, 'Ausente', 6, '2026-05-05');
150
151 •   INSERT INTO Reportes (nombre_reporte, tipo_reporte, fecha_generacion, ruta_archivo, id_usuario, parametros) VALUES
152     ('Reporte de Alertas Séptimo A', 'PDF', '2026-05-10', '/outputs/pdf/alertas_7a.pdf', 6, 'grado=1'),
153     ('Uso Indebido de Celulares Octavo', 'Excel', '2026-05-11', '/outputs/excel/celulares_8.xlsx', 9, 'habito=4'),
154     ('Gráfico de Rendimiento Undécimo', 'Grafico', '2026-05-11', '/outputs/img/promo_2026.png', 1, 'mes=mayo'),
155     ('Reporte Psicoeducativo Juan Alvarez', 'PDF', '2026-05-12', '/outputs/pdf/rep_juan.pdf', 6, 'estudiante=1'),
156     ('Consolidado de Tareas Noveno', 'Excel', '2026-05-12', '/outputs/excel/tareas_9.xlsx', 4, 'grado=3'),
157     ('Gráfico Asistencia Décimo A', 'Grafico', '2026-05-13', '/outputs/img/asistencia_10.png', 2, 'grado=4'),
158     ('Bitácora de Disciplina General', 'PDF', '2026-05-13', '/outputs/pdf/disciplina.pdf', 9, 'tipo=alerta'),
159     ('Alumnos en Tutorías de Matemáticas', 'Excel', '2026-05-13', '/outputs/excel/tutorias.xlsx', 4, 'materia=mate'),
160     ('Reporte Individual Rosa Espinoza', 'PDF', '2026-05-13', '/outputs/pdf/rep_rosa.pdf', 5, 'estudiante=5'),
161     ('Estadísticas de Convivencia Escolar', 'Grafico', '2026-05-13', '/outputs/img/convivencia.png', 6, 'periodo=parcial1'
```

5.4. Tablas con datos reales

```
163     SELECT * FROM Grado;
164 •   SELECT * FROM Usuario;
165 •   SELECT * FROM Habito;
166 •   SELECT * FROM Seccion;
167 •   SELECT * FROM Estudiante;
168 •   SELECT * FROM Registro;
169 •   SELECT * FROM Reportes;
170
171
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_grado	nombre_grado	nivel
1	Séptimo Grado	Secundaria
2	Octavo Grado	Secundaria
3	Noveno Grado	Secundaria
4	Décimo Grado	Secundaria
5	Undécimo Grado	Secundaria
6	Séptimo Grado B	Secundaria
7	Octavo Grado B	Secundaria
8	Noveno Grado B	Secundaria
9	Décimo Grado B	Secundaria
10	Undécimo Grado B	Secundaria

```
163     SELECT * FROM Grado;
164 •   SELECT * FROM Usuario;
165 •   SELECT * FROM Habito;
166 •   SELECT * FROM Seccion;
167 •   SELECT * FROM Estudiante;
168 •   SELECT * FROM Registro;
169 •   SELECT * FROM Reportes;
170
171
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_usuario	nombre	apellido	email	contrasena	tipo_usuario	estado
1	Ana	Martinez	ana.mtz@colegio.com	pass123	Administrador	1
2	Carlos	Gomez	carlos.g@colegio.com	secure456	Docente	1
3	Lucia	Lopez	lucia.l@colegio.com	admin789	Administrador	1
4	Jorge	Rodriguez	jorge.r@colegio.com	clavedocente	Docente	1
5	Elena	Perez	elena.p@colegio.com	perez2026	Docente	1
6	David	Sánchez	david.s@colegio.com	david99	Docente	1
7	Sofia	Diaz	sofia.d@colegio.com	sofia123	Administrador	1
8	Miguel	Torres	miguel.t@colegio.com	torres456	Docente	1
9	Laura	Castro	laura.c@colegio.com	laura789	Administrador	1
10	Pedro	Morales	pedro.m@colegio.com	pedro2026	Docente	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL


```
163     SELECT * FROM Grado;
164 •   SELECT * FROM Usuario;
165 •   SELECT * FROM Habito;
166 •   SELECT * FROM Seccion;
167 •   SELECT * FROM Estudiante;
168 •   SELECT * FROM Registro;
169 •   SELECT * FROM Reportes;
170
171
```


Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_habito	nombre_habito	tipo_habito	unidad_medida	estado
1	Entrega de Ensayos	Académico	Porcentaje	Activo
2	Puntualidad en Bloque 1	Conducta	Minutos tarde	Activo
3	Participación en Debates	Académico	Interacciones	Activo
4	Uso Correcto de Celular	Disciplina	Llamadas de atención	Activo
5	Trabajo de Laboratorio	Académico	Calificación	Activo
6	Asistencia a Tutorías	Apoyo	Horas asistidas	Activo
7	Respeto en Comunidad	Convivencia	Incidentes	Activo
8	Cuidado del Mobiliario	Disciplina	Reportes de daño	Activo
9	Liderazgo en Proyectos	Social	Puntos acumulados	Activo
10	Uso de Uniforme Completo	Disciplina	Cumplimiento	Activo

```
163     SELECT * FROM Grado;
164 •   SELECT * FROM Usuario;
165 •   SELECT * FROM Habito;
166 •   SELECT * FROM Seccion;
167 •   SELECT * FROM Estudiante;
168 •   SELECT * FROM Registro;
169 •   SELECT * FROM Reportes;
170
171
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_seccion	nombre_seccion	id_grado
1	Única	1
2	Sección A	2
3	Sección B	2
4	Única	3
5	Sección A	4
6	Sección B	4
7	Matutina	5
8	Vespertina	5
9	Sección A	6
10	Única	7

```
163     SELECT * FROM Grado;
164 •   SELECT * FROM Usuario;
165 •   SELECT * FROM Habito;
166 •   SELECT * FROM Seccion;
167 •   SELECT * FROM Estudiante;
168 •   SELECT * FROM Registro;
169 •   SELECT * FROM Reportes;
170
171
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_estudiante	nombre	apellido	email	fecha_nacimiento	id_grado
1	Juan	Alvarez	juan.alvarez@secundaria.com	2014-04-12	1
2	Maria	Benitez	maria.benitez@secundaria.com	2013-08-22	2
3	Luis	Cabrera	luis.cabrera@secundaria.com	2012-01-15	3
4	Andres	Delgado	andres.delgado@secundaria.com	2011-11-05	4
5	Rosa	Espinoza	rosa.espinoza@secundaria.com	2010-06-30	5
6	Kevin	Flores	kevin.flores@secundaria.com	2013-02-14	6
7	Camila	Gutierrez	camila.gutierrez@secundaria.com	2012-09-18	7
8	Diego	Herrera	diego.herrera@secundaria.com	2011-05-25	8
9	Natalia	Ibarra	natalia.ibarra@secundaria.com	2010-12-01	9
10	Gabriel	Juarez	gabriel.juarez@secundaria.com	2009-07-07	10

```
163     SELECT * FROM Grado;
164 •   SELECT * FROM Usuario;
165 •   SELECT * FROM Habito;
166 •   SELECT * FROM Seccion;
167 •   SELECT * FROM Estudiante;
168 •   SELECT * FROM Registro;
169 •   SELECT * FROM Reportes;
170
171
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_registro	id_estudiante	id_habito	fecha_registro	valor	observacion	id_usuario	fecha_creacion
1	1	1	2026-05-01	0	Ausente	2	2026-05-01
2	2	4	2026-05-01	0	No cumplió en nada	6	2026-05-01
3	3	2	2026-05-02	15	Bajo rendimiento	4	2026-05-02
4	4	3	2026-05-02	0	Ausente	4	2026-05-02
5	5	5	2026-05-03	0	No cumplió en nada	2	2026-05-03
6	6	1	2026-05-03	4	Bajo rendimiento	5	2026-05-03
7	7	10	2026-05-04	0	Ausente	6	2026-05-04
8	8	7	2026-05-04	3	No cumplió en nada	10	2026-05-04
9	9	2	2026-05-05	25	Bajo rendimiento	4	2026-05-05
10	10	4	2026-05-05	0	Ausente	6	2026-05-05

```
163     SELECT * FROM Grado;
164 •   SELECT * FROM Usuario;
165 •   SELECT * FROM Habito;
166 •   SELECT * FROM Seccion;
167 •   SELECT * FROM Estudiante;
168 •   SELECT * FROM Registro;
169 •   SELECT * FROM Reportes;
170
171
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_reporte	nombre_reporte	tipo_reporte	fecha_generacion	ruta_archivo	id_usuario	parametros
1	Reporte de Alertas Séptimo A	PDF	2026-05-10	/outputs/pdf/alertas_7a.pdf	6	grado=1
2	Uso Indebido de Celulares Octavo	Excel	2026-05-11	/outputs/excel/celulares_8.xlsx	9	habito=4
3	Gráfico de Rendimiento Undécimo	Grafico	2026-05-11	/outputs/img/promo_2026.png	1	mes=mayo
4	Reporte Psicoeducativo Juan Alvarez	PDF	2026-05-12	/outputs/pdf/rep_juan.pdf	6	estudiante=1
5	Consolidado de Tareas Noveno	Excel	2026-05-12	/outputs/excel/tareas_9.xlsx	4	grado=3
6	Gráfico Asistencia Décimo A	Grafico	2026-05-13	/outputs/img/asistencia_10.png	2	grado=4
7	Bitácora de Disciplina General	PDF	2026-05-13	/outputs/pdf/disciplina.pdf	9	tipo=alerta
8	Alumnos en Tutorías de Matemáticas	Excel	2026-05-13	/outputs/excel/tutorias.xlsx	4	materia=mate
9	Reporte Individual Rosa Espinoza	PDF	2026-05-13	/outputs/pdf/rep_rosa.pdf	5	estudiante=5
10	Estadísticas de Convivencia Escolar	Grafico	2026-05-13	/outputs/img/convivencia.png	6	periodo=parcial1
NULL	NULL	NULL	NULL	NULL	NULL	NULL

6. CONSULTAS DISEÑADAS A LA BASE DE DATOS

```sql
USE SistemaRegistroHabitos;

SELECT
    e.nombre AS Estudiante,
    e.apellido AS Apellido,
    g.nombre_grado AS Grado,
    h.nombre_habito AS Indicador,
    r.observacion AS Estado
FROM Registro r
INNER JOIN Estudiante e ON r.id_estudiante = e.id_estudiante
INNER JOIN Grado g ON e.id_grado = g.id_grado
INNER JOIN Habito h ON r.id_habito = h.id_habito
WHERE r.observacion IN ('Ausente', 'No cumplió en nada', 'Bajo rendimiento');

UPDATE Registro
SET valor = 4.5, observacion = 'Mejoró rendimiento'
WHERE id_estudiante = 3 AND id_habito = 2;

SELECT * FROM Registro;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_registro	id_estudiante	id_habito	fecha_registro	valor	observacion	id_usuario	fecha_creacion
1	1	1	2026-05-01	0	Ausente	2	2026-05-01
2	2	4	2026-05-01	0	No cumplió en nada	6	2026-05-01
3	3	2	2026-05-02	4.5	Mejoró rendimiento	4	2026-05-02
4	4	3	2026-05-02	0	Ausente	4	2026-05-02
5	5	5	2026-05-03	0	No cumplió en nada	2	2026-05-03
6	6	1	2026-05-03	4	Bajo rendimiento	5	2026-05-03
7	7	10	2026-05-04	0	Ausente	6	2026-05-04
8	8	7	2026-05-04	3	No cumplió en nada	10	2026-05-04
9	9	2	2026-05-05	25	Bajo rendimiento	4	2026-05-05
10	10	4	2026-05-05	0	Ausente	6	2026-05-05
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
24 •  DELETE FROM Reportes
25    WHERE id_reporte = 10;
26
27 •  SELECT * FROM Reportes;
28
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

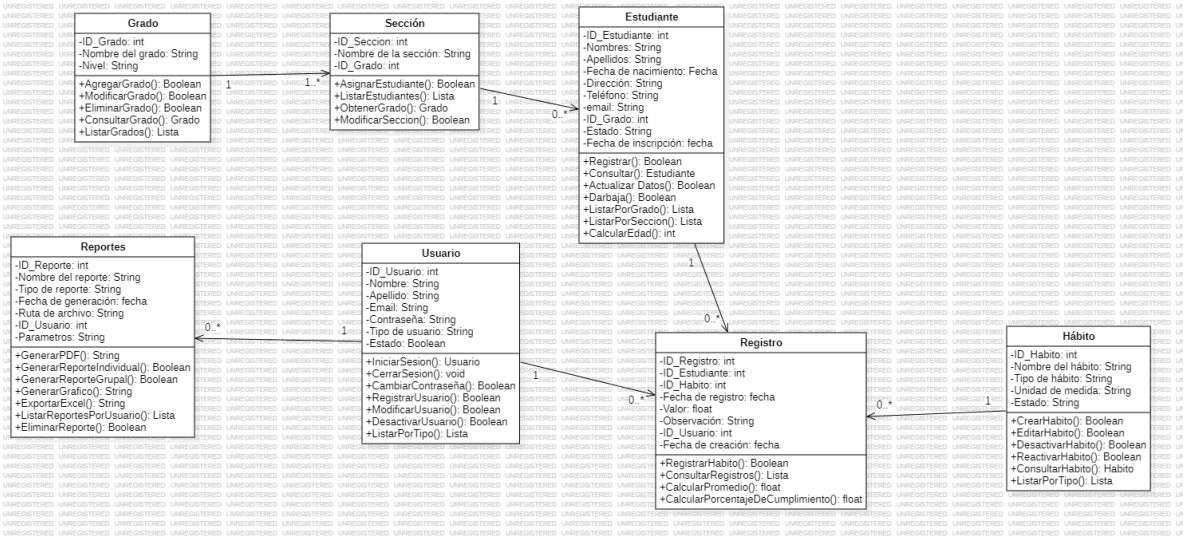
id_reporte	nombre_reporte	tipo_reporte	fecha_generacion	ruta_archivo	id_usuario	parametros
1	Reporte de Alertas Séptimo A	PDF	2026-05-10	/outputs/pdf/alertas_7a.pdf	6	grado=1
2	Uso Indebido de Celulares Octavo	Excel	2026-05-11	/outputs/excel/celulares_8.xlsx	9	habito=4
3	Gráfico de Rendimiento Undécimo	Grafico	2026-05-11	/outputs/img/promo_2026.png	1	mes=mayo
4	Reporte Psicoeducativo Juan Alvarez	PDF	2026-05-12	/outputs/pdf/rep_juan.pdf	6	estudiante=1
5	Consolidado de Tareas Noveno	Excel	2026-05-12	/outputs/excel/tareas_9.xlsx	4	grado=3
6	Gráfico Asistencia Décimo A	Grafico	2026-05-13	/outputs/img/asistencia_10.png	2	grado=4
7	Bitácora de Disciplina General	PDF	2026-05-13	/outputs/pdf/disciplina.pdf	9	tipo=alerta
8	Alumnos en Tutorías de Matemáticas	Excel	2026-05-13	/outputs/excel/tutorias.xlsx	4	materia=mate
9	Reporte Individual Rosa Espinoza	PDF	2026-05-13	/outputs/pdf/rep_rosa.pdf	5	estudiante=5
NULL	NULL	NULL	NULL	NULL	NULL	NULL

7. DISEÑO DE ARQUITECTURA DE APLICACIÓN

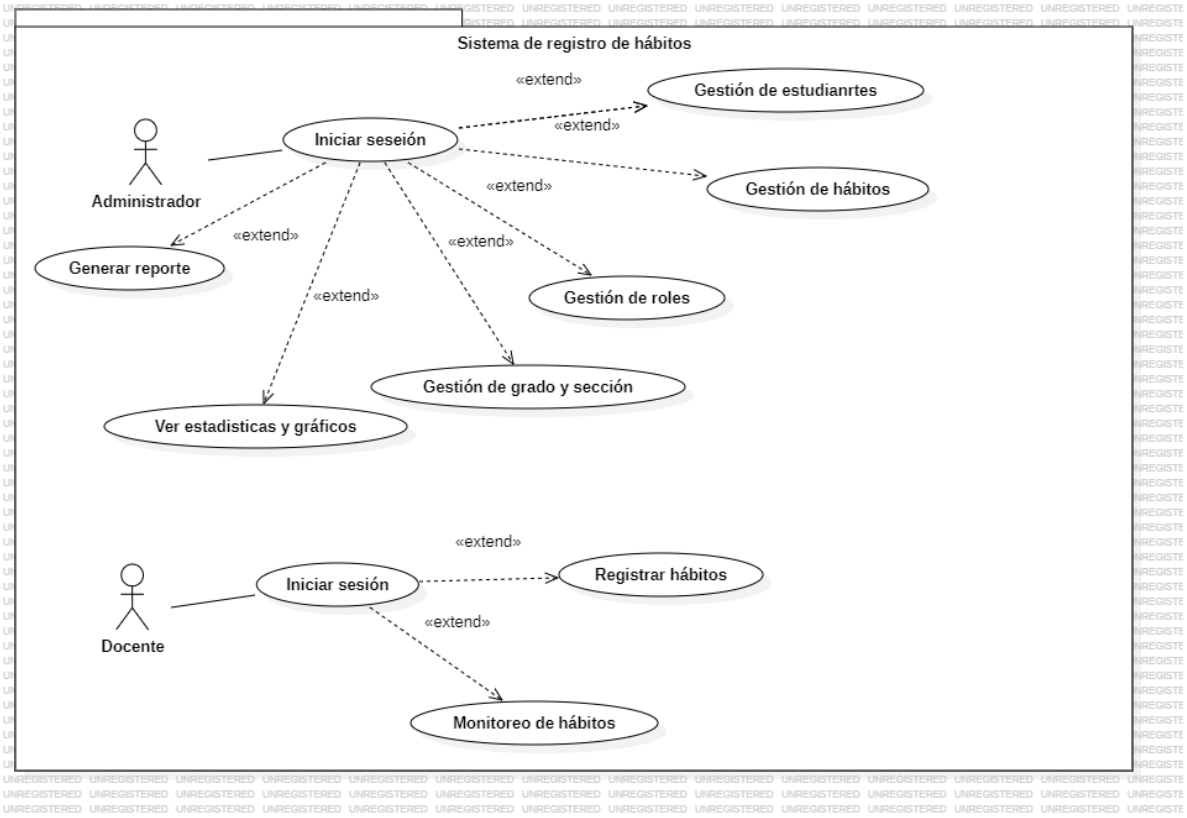
Capa	**Función**	**Componentes del sistema**
Capa de presentación	Interfaz que ve el usuario	Pantallas diseñadas en figma
Capa de Negocio	Reglas y cálculos del sistema	Clases y métodos
Capa de Datos	Almacenamiento de información	MySQL, tablas, consultas SQL

8. DIAGRAMAS UML

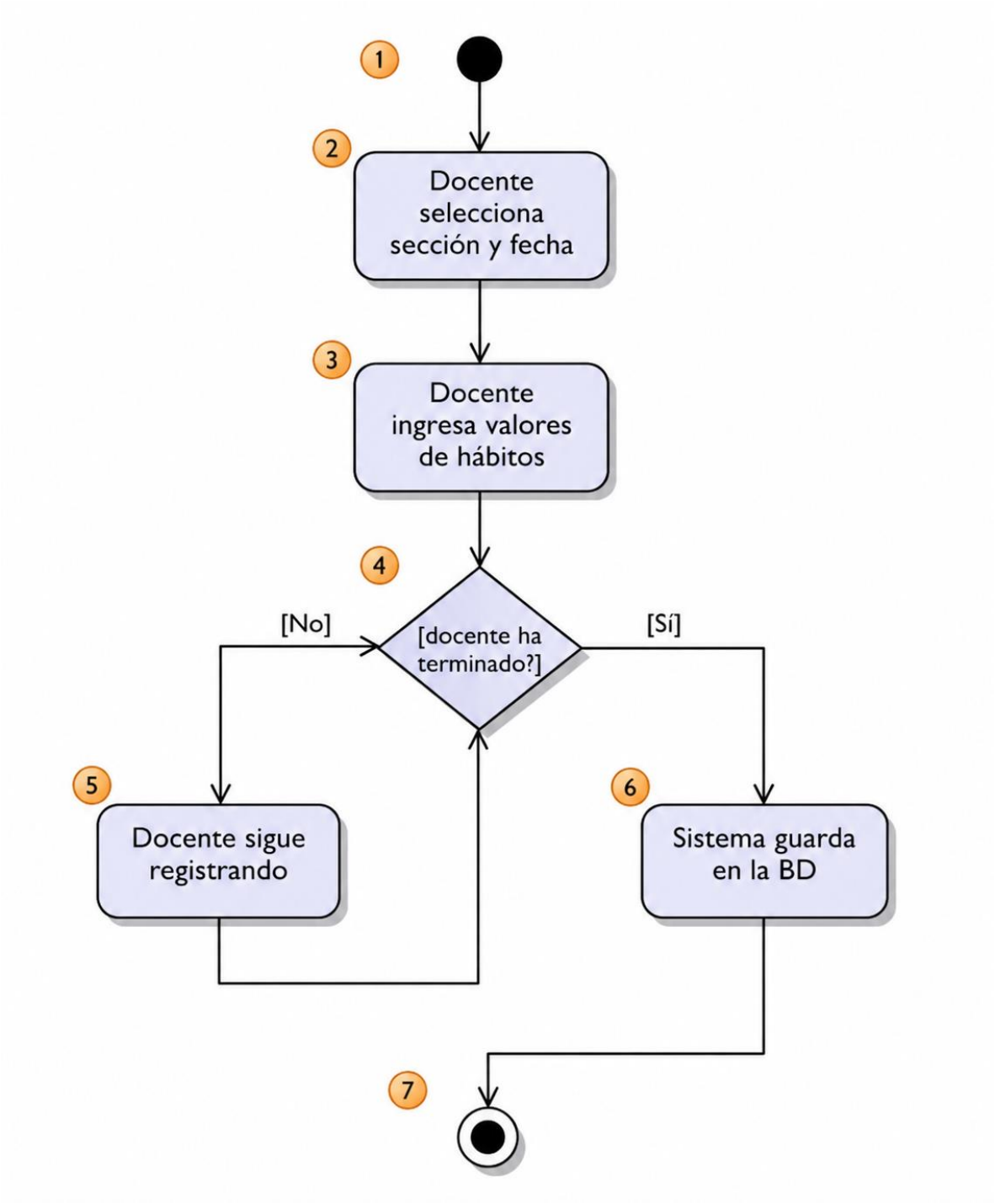
8.1. Diagrama de clases



8.2. Diagrama de caso de uso



8.3. Diagrama de actividades



9. CODIGO DE CLASES DE ENTIDAD (CAPAS DE DATOS)

9.1. Clase Grado

```java
package Modelo;
public class Grado {
    private int idGrado;
    private String nombreGrado;
    private String nivel;

    public Grado() {}
    public Grado(int idGrado, String nombreGrado, String nivel) {
        this.idGrado = idGrado;
        this.nombreGrado = nombreGrado;
        this.nivel = nivel;
    }
    public int getIdGrado() {
        return idGrado;
    }

    public void setIdGrado(int idGrado) {
        this.idGrado = idGrado;
    }

    public String getNombreGrado() {
        return nombreGrado;
    }

    public void setNombreGrado(String nombreGrado) {
        this.nombreGrado = nombreGrado;
    }

    public String getNivel() {
        return nivel;
    }

    public void setNivel(String nivel) {
        this.nivel = nivel;
    }
}
```

9.2. Clase Sección

```
2   package Modelo;
3
4
5
6   public class Sección {
7       private int idSeccion;
8       private String nombreSeccion;
9       private int idGrado;
10
11      public Sección() {
12      }
13
14      public Sección(int idSeccion, String nombreSeccion, int idGrado) {
15          this.idSeccion = idSeccion;
16          this.nombreSeccion = nombreSeccion;
17          this.idGrado = idGrado;
18      }
19
20      public int getIdSeccion() {
21          return idSeccion;
22      }
23
24      public void setIdSeccion(int idSeccion) {
25          this.idSeccion = idSeccion;
26      }
27
28      public String getNombreSeccion() {
29          return nombreSeccion;
30      }
```

```
32      public void setNombreSeccion(String nombreSeccion) {
33          this.nombreSeccion = nombreSeccion;
34      }
35
36      public int getIdGrado() {
37          return idGrado;
38      }
39
40      public void setIdGrado(int idGrado) {
41          this.idGrado = idGrado;
42      }
43
44
45
46  }
47
48
```

9.3. Clase Estudiante

```java
package Modelo;

import java.time.LocalDate;


public class Estudiante {
    // Atributos
     private int idEstudiante;
    private String nombre;
    private String apellido;
    private String email;
    private LocalDate fechaNacimiento;
    private int idGrado;

    public Estudiante() {
    }



    public Estudiante(int idEstudiante, String nombre, String apellido, String email, LocalDate fechaNacimiento, int idGrado) {
        this.idEstudiante = idEstudiante;
        this.nombre = nombre;
        this.apellido = apellido;
        this.email = email;
        this.fechaNacimiento = fechaNacimiento;
        this.idGrado = idGrado;
    }
```

```java

    public int getIdEstudiante() {
        return idEstudiante;
    }

    public void setIdEstudiante(int idEstudiante) {
        this.idEstudiante = idEstudiante;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public String getApellido() {
        return apellido;
    }

    public void setApellido(String apellido) {
        this.apellido = apellido;
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }
```

```
61
62      public LocalDate getFechaNacimiento() {
63          return fechaNacimiento;
64      }
65
66      public void setFechaNacimiento(LocalDate fechaNacimiento) {
67          this.fechaNacimiento = fechaNacimiento;
68      }
69
70      public int getIdGrado() {
71          return idGrado;
72      }
73
74      public void setIdGrado(int idGrado) {
75          this.idGrado = idGrado;
76      }}
77
78
79
```

9.4. Clase Hábito

```
1
2   package Modelo;
3
4   public class Hábito {
5
6       private int idHabito;
7       private String nombreHabito;
8       private String tipoHabito;
9       private String unidadMedida;
10      private String estado;
11
12      public Hábito() {
13      }
14
15      public Hábito(int idHabito, String nombreHabito, String tipoHabito, String unidadMedida, String estado) {
16          this.idHabito = idHabito;
17          this.nombreHabito = nombreHabito;
18          this.tipoHabito = tipoHabito;
19          this.unidadMedida = unidadMedida;
20          this.estado = estado;
21      }
22
23      public int getIdHabito() {
24          return idHabito;
25      }
26
27      public void setIdHabito(int idHabito) {
28          this.idHabito = idHabito;
29      }
```

```java
31      public String getNombreHabito() {
32          return nombreHabito;
33      }
34
35      public void setNombreHabito(String nombreHabito) {
36          this.nombreHabito = nombreHabito;
37      }
38
39      public String getTipoHabito() {
40          return tipoHabito;
41      }
42
43      public void setTipoHabito(String tipoHabito) {
44          this.tipoHabito = tipoHabito;
45      }
46
47      public String getUnidadMedida() {
48          return unidadMedida;
49      }
50
51      public void setUnidadMedida(String unidadMedida) {
52          this.unidadMedida = unidadMedida;
53      }
54
55      public String getEstado() {
56          return estado;
57      }
58
59      public void setEstado(String estado) {
60          this.estado = estado;
61      }
```

9.5. Clase Usuario

```java
1
2   package Modelo;
3
4   public class Usuario {
5
6       private int idUsuario;
7       private String nombre;
8       private String apellido;
9       private String email;
10      private String contrasena;
11      private String tipoUsuario;
12      private boolean estado;
13
14      public Usuario() {
15      }
16
17      public Usuario(int idUsuario, String nombre, String apellido, String email, String contrasena, String tipoUsuario, boolean estado) {
18          this.idUsuario = idUsuario;
19          this.nombre = nombre;
20          this.apellido = apellido;
21          this.email = email;
22          this.contrasena = contrasena;
23          this.tipoUsuario = tipoUsuario;
24          this.estado = estado;
25      }
26
27      public int getIdUsuario() {
28          return idUsuario;
29      }
```

```
31      public void setIdUsuario(int idUsuario) {
32          this.idUsuario = idUsuario;
33      }
34
35      public String getNombre() {
36          return nombre;
37      }
38
39      public void setNombre(String nombre) {
40          this.nombre = nombre;
41      }
42
43      public String getApellido() {
44          return apellido;
45      }
46
47      public void setApellido(String apellido) {
48          this.apellido = apellido;
49      }
50
51      public String getEmail() {
52          return email;
53      }
54
55      public void setEmail(String email) {
56          this.email = email;
57      }
58
59      public String getContrasena() {
60          return contrasena;
61      }
```

```java
    }

    public void setContrasena(String contrasena) {
        this.contrasena = contrasena;
    }

    public String getTipoUsuario() {
        return tipoUsuario;
    }

    public void setTipoUsuario(String tipoUsuario) {
        this.tipoUsuario = tipoUsuario;
    }

    public boolean isEstado() {
        return estado;
    }

    public void setEstado(boolean estado) {
        this.estado = estado;
    }
```

9.6. Clase Registro

```java
package Modelo;

import java.time.LocalDate;



public class Registro {
    private int idRegistro;
    private int idEstudiante;
    private int idHabito;
    private LocalDate fechaRegistro;
    private float valor;
    private String observacion;
    private int idUsuario;
    private LocalDate fechaCreacion;

    public Registro() {
    }

    public Registro(int idRegistro, int idEstudiante, int idHabito, LocalDate fechaRegistro, float valor, String observacion, int idUsuario,
            LocalDate fechaCreacion) {
        this.idRegistro = idRegistro;
        this.idEstudiante = idEstudiante;
        this.idHabito = idHabito;
        this.fechaRegistro = fechaRegistro;
        this.valor = valor;
        this.observacion = observacion;
        this.idUsuario = idUsuario;
        this.fechaCreacion = fechaCreacion;
```

```java
34      public int getIdRegistro() {
35          return idRegistro;
36      }
37
38      public void setIdRegistro(int idRegistro) {
39          this.idRegistro = idRegistro;
40      }
41
42      public int getIdEstudiante() {
43          return idEstudiante;
44      }
45
46      public void setIdEstudiante(int idEstudiante) {
47          this.idEstudiante = idEstudiante;
48      }
49
50      public int getIdHabito() {
51          return idHabito;
52      }
53
54      public void setIdHabito(int idHabito) {
55          this.idHabito = idHabito;
56      }
57
58      public LocalDate getFechaRegistro() {
59          return fechaRegistro;
60      }
61
62      public void setFechaRegistro(LocalDate fechaRegistro) {
63          this.fechaRegistro = fechaRegistro;
64      }
```

```java
66      public float getValor() {
67          return valor;
68      }
69
70      public void setValor(float valor) {
71          this.valor = valor;
72      }
73
74      public String getObservacion() {
75          return observacion;
76      }
77
78      public void setObservacion(String observacion) {
79          this.observacion = observacion;
80      }
81
82      public int getIdUsuario() {
83          return idUsuario;
84      }
85
86      public void setIdUsuario(int idUsuario) {
87          this.idUsuario = idUsuario;
88      }
89
90      public LocalDate getFechaCreacion() {
91          return fechaCreacion;
92      }
93
94      public void setFechaCreacion(LocalDate fechaCreacion) {
95          this.fechaCreacion = fechaCreacion;
96      }
```

9.7. Clase Reporte

```java
2   package Modelo;
3
4   import java.time.LocalDate;
5
6
7   public class Reporte {
8
9       private int idReporte;
10      private String nombreReporte;
11      private String tipoReporte;
12      private LocalDate fechaGeneracion;
13      private String rutaArchivo;
14      private int idUsuario;
15      private String parametros;
16
17      public Reporte() {
18      }
19
        public Reporte(int idReporte, String nombreReporte, String tipoReporte, LocalDate fechaGeneracion, String rutaArchivo,
21              int idUsuario, String parametros) {
22          this.idReporte = idReporte;
23          this.nombreReporte = nombreReporte;
24          this.tipoReporte = tipoReporte;
25          this.fechaGeneracion = fechaGeneracion;
26          this.rutaArchivo = rutaArchivo;
27          this.idUsuario = idUsuario;
28          this.parametros = parametros;
29      }
```

```java
31      public int getIdReporte() {
32          return idReporte;
33      }
34
35      public void setIdReporte(int idReporte) {
36          this.idReporte = idReporte;
37      }
38
39      public String getNombreReporte() {
40          return nombreReporte;
41      }
42
43      public void setNombreReporte(String nombreReporte) {
44          this.nombreReporte = nombreReporte;
45      }
46
47      public String getTipoReporte() {
48          return tipoReporte;
49      }
50
51      public void setTipoReporte(String tipoReporte) {
52          this.tipoReporte = tipoReporte;
53      }
54
55      public LocalDate getFechaGeneracion() {
56          return fechaGeneracion;
57      }
58
59      public void setFechaGeneracion(LocalDate fechaGeneracion) {
60          this.fechaGeneracion = fechaGeneracion;
61      }
```

```
62
63      public String getRutaArchivo() {
64          return rutaArchivo;
65      }
66
67      public void setRutaArchivo(String rutaArchivo) {
68          this.rutaArchivo = rutaArchivo;
69      }
70
71      public int getIdUsuario() {
72          return idUsuario;
73      }
74
75      public void setIdUsuario(int idUsuario) {
76          this.idUsuario = idUsuario;
77      }
78
79      public String getParametros() {
80          return parametros;
81      }
82
83      public void setParametros(String parametros) {
84          this.parametros = parametros;
85      }
86
87
```

10.DISEÑO FUNCIONAL DEL PROTOTIPO (ENLACE FIGMA)

https://www.figma.com/design/j8jnrwrKCJxfF8sxvYNzSP/Grupo-4?node-id=286-120&t=XgQXznXaTFjsjv0C-1

11. IMPLEMENTACIÓN DE OPERACIONES CRUD (CÓDIGO DOCUMENTADAS)

11.1. GradoDAO

```java
package Modelo;

import Modelo.Grado;
import java.util.ArrayList;
import java.util.List;


public class GradoDAO {

    public boolean agregarGrado(Grado g) { return true; }
    public boolean modificarGrado(Grado g) { return true; }
    public boolean eliminarGrado(int id) { return true; }
    public Grado consultarGrado(int id) { return new Grado(); }
    public List<Grado> listarGrados() { return new ArrayList<>(); }
}
```

11.2. SecciónDAO

```java
package Modelo;

import Modelo.Sección;
import java.util.ArrayList;
import java.util.List;

public class SecciónDAO {

    public boolean asignarEstudiante(int idEstudiante, int idSeccion) { return true; }
    public List<Estudiante> listarEstudiantes(int idSeccion) { return new ArrayList<>(); }
    public Grado obtenerGrado(int idSeccion) { return new Grado(); }
    public boolean modificarSeccion(Sección s) { return true; }
}
```

11.3. EstudianteDAO

```java
package Modelo;


import Modelo.Estudiante;
import java.util.ArrayList;
import java.util.List;


public class EstudianteDAO {
    public boolean agregarEstudiante(Estudiante e) { return true; }
    public boolean modificarEstudiante(Estudiante e) { return true; }
    public boolean eliminarEstudiante(int id) { return true; }
    public Estudiante consultarEstudiante(int id) { return new Estudiante(); }
    public List<Estudiante> listarEstudiantes() { return new ArrayList<>(); }
}
```

11.4. HábitoDAO

```java

package Modelo;

import Modelo.Hábito;
import java.util.ArrayList;
import java.util.List;


public class HábitoDAO {
    public boolean crearHabito(Hábito h) { return true; }
    public boolean editarHabito(Hábito h) { return true; }
    public boolean desactivarHabito(int id) { return true; }
    public boolean reactivarHabito(int id) { return true; }
    public Hábito consultarHabito(int id) { return new Hábito(); }
    public List<Hábito> listarPorTipo(String tipo) { return new ArrayList<>(); }
}
```

11.5. UsuarioDAO

```java

package Modelo;

import Modelo.Usuario;
import java.util.ArrayList;
import java.util.List;


public class UsuarioDAO {


    public Usuario iniciarSesion(String email, String contrasena) { return new Usuario(); }
    public void cerrarSesion() {}
    public boolean cambiarContrasena(int idUsuario, String nuevaContra) { return true; }
    public boolean registrarUsuario(Usuario u) { return true; }
    public boolean modificarUsuario(Usuario u) { return true; }
    public boolean desactivarUsuario(int id) { return true; }
    public List<Usuario> listarPorTipo(String tipo) { return new ArrayList<>(); }

}
```

11.6. RegistroDAO

```java

package Modelo;

import Modelo.Registro;
import java.util.ArrayList;
import java.util.List;


public class RegistroDAO {
    public boolean registrarHabito(Registro r) { return true; }
    public List<Registro> consultarRegistros() { return new ArrayList<>(); }
    public float calcularPromedio(int idEstudiante, int idHabito) { return 0.0f; }
    public float calcularPorcentajeDeCumplimiento(int idEstudiante, int idHabito) { return 100.0f; }

}

```

11.7. ReporteDAO

```java

package Modelo;

import Modelo.Reporte;
import java.util.ArrayList;
import java.util.List;


public class ReporteDAO {
    public String generarPDF(Reporte r) { return "Ruta/archivo.pdf"; }
    public boolean generarReporteIndividual() { return true; }
    public boolean generarReporteGrupal() { return true; }
    public String generarGrafico() { return "GraficoBase64"; }
    public String exportarExcel(Reporte r) { return "Ruta/archivo.xlsx"; }
    public List<Reporte> listarReportesPorUsuario(int idUsuario) { return new ArrayList<>(); }
    public boolean eliminarReporte(int idReporte) { return true; }
}

```

12.REFERENCIAS BIBLIOGRÁFICAS

Carazas Durand, C. R., Garcia Soller, T. M., Barrios Sánchez, F., Florez Cueva, M. Y., & Galecio Mora, D. E. (2026). Factores sociodemográficos vinculados a las habilidades sociales en estudiantes de educación básica. Revista InveCom, 6(1). https://doi.org/10.5281/zenodo.15376586

Gobierno de Reconciliación y Unidad Nacional. (2025). Estrategia Escuela Saludable 2025. Ministerio de Salud (MINSA) y Ministerio de Educación (MINED), Nicaragua.

Gutiérrez, I. (2025). Transformando la educación: Nicaragua inicia estrategia saludable 2025. Barricada.
https://www.diariobarricada.com

Hernández-Beltrán, V., Gámez-Calvo, L., & Muñoz-Jiménez, E. (2022). Revisión sistemática de programas de intervención para promover hábitos saludables de actividad física y nutrición en escolares españoles. Archivos Latinoamericanos de Nutrición, 72(4), 294-305.

Lobach, Y., Suárez, A. J. G., Berlanga, M. D. C. L., Tiomno, K. S., Zebede, J. A., & Cordero, J. C. G. (2024). Hábitos saludables y conductas de riesgo en una muestra de adolescentes de Nicaragua. Journal of Physical Education and Human Movement, 6(1), 4-13. https://dialnet.unirioja.es/servlet/articulo?codigo=XXXXX

Ministerio de Educación de Nicaragua. (2024). Estrategia Nacional de Educación en todas sus Modalidades "Bendiciones y Victorias" 2024-2026. Managua: MINED.

Organización Mundial de la Salud. (2023). Promoción de la salud en el entorno escolar. Ginebra: OMS.

Pressman, R. S. (2021). Ingeniería de software: Un enfoque práctico (9ª ed.). México: McGraw-Hill.

UNAN-Managua. (2024). Hábitos de alimentación saludable son fomentados desde la educación primaria. Reportajes UNAN-Managua

https://www.unan.edu.ni

Viñamagua Chicaiza, C. E., Lucas Reyes, J. J., & Cevallos Suarez, S. E. (2026). Estrategias para abordar los determinantes socioculturales del embarazo adolescente en Ecuador. Prohominum, 8(1).

https://doi.org/10.47606/acven/ph0427

13. ANEXOS

13.1. Informe de sesiones de trabajos

Sesión #1:	**Definición del tema**	**Fecha / Hora:**	**07-03-26**
N°	**Nombres y apellidos**	**Aportes realizados por integrantes de equipo**	**Observaciones**
1	Nerybeth Auxiliadora Bravo López	Aportó en todo	
2	Anderson Johan Gonzalez Torrez	Aportó en todo	
3	Nathali Belén Centeno Miranda	Aportó en todo	
4	Pablo Francisco Bravo Centeno	Aportó en todo	
5	Yeison Alexander Gomez Morales	Aportó en todo	

Sesión #2:	**Revisión de Scrum y creación de Diagramas de procesos y UML**	**Fecha / Hora:**	**05-05-26**
N°	**Nombres y apellidos**	**Aportes realizados por integrantes de equipo**	**Observaciones**
1	Nerybeth Auxiliadora Bravo López	Aportó en todo	
2	Anderson Johan Gonzalez Torrez	Aportó en todo	
3	Pablo Francisco Bravo Centeno	Aportó en todo	
4	Yeison Alexander Gomez Morales	Aportó en todo	

Sesión #3:	**Diseño de BD y avance de programación**	**Fecha / Hora:**	**07-05-26**
N°	**Nombres y apellidos**	**Aportes realizados por integrantes de equipo**	**Observaciones**
1	Nerybeth Auxiliadora Bravo López	Aportó en todo	

2	Anderson Johan Gonzalez Torrez	Aportó en todo	
3	Pablo Francisco Bravo Centeno	Aportó en todo	
4	Yeison Alexander Gomez Morales	Aportó en todo	

13.2. Cronograma

N°	**Actividades/Tareas**	**Fecha inicio**	**Fecha fin**	**Hrs de trabajo**	**Responsable de actividad**
1	Selección de tema	07-03-26	07-03-26	1 hr	Equipo
2	Elaboración del documento	16-03-26	17-04-26	48hr	Equipo
3	Diseño de prototipo	21-03-26	15-05-26	48hr	Equipo
4	Elaboración de encuesta	16-04-26	17-04-26	5hr	Equipo
4	Correcciones de primer corte	21-04-26	25-04-26	12hr	Equipo
5	Base de datos	07-05-26	14-05-26	48hr	Equipo
6	Mapeo de procesos	25-04-26	09-05-26	48hr	Equipo
7	Diagramas UML	25-04-26	13-05-26	48hr	Equipo
8	Programación	07-05-26	15-05-26	72hr	Equipo