



**TÍTULO:**

App de recordatorio de medicamentos y seguimiento de tratamientos

**CURSO:**

Análisis y Diseño de Software

**DOCENTE:**

Osorio Contreras, Rosario Delia

**INTEGRANTES:**

Castillo Lindo, Daniel Ricardo

Mendez Herrera, Cristopher Marti

Ruiz Huapaya, Tatiana Carolay

Zevallos Jara, Edgar Franco

**LINK DE PRESENTACIÓN:**

<https://continental-team-qbitwq07.atlassian.net/jira/software/projects/MED/boards/34/backlog?epics=visible&atlOrigin=eyJpIjoiMDIxMTA3Y2lwOTNiNDMwOWExZGVkMWU5ZTZmYTZhYzciLCJwIjoiaj9>

[https://www.canva.com/design/DAG0eMGA4Ls/bhWuK\\_RF-xVD6uXgbWDmAQ/edit?utm\\_content=DAG0eMGA4Ls&utm\\_campaign=designshare&utm\\_medium=link2&utm\\_source=sharebutton](https://www.canva.com/design/DAG0eMGA4Ls/bhWuK_RF-xVD6uXgbWDmAQ/edit?utm_content=DAG0eMGA4Ls&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)

**HUANCAYO – PERÚ**

**2025**

## Índice

Resumen.....	3
Introducción.....	4
1. Capítulo 1-Presentación del Proyecto.....	5
2. Capítulo 2-Análisis de Necesidades y Requerimientos.....	5
3. Capítulo 3-Modelos Iniciales del Sistema.....	5
4. Capítulo 4- Modelos de Diseño.....	6
5. Capítulo 5-Metodología de Trabajo (SCRUM).....	6
6. Conclusiones y Recomendaciones.....	6
7. Referencias Bibliográficas.....	6
8. Anexos.....	6

## **Resumen**

El presente informe desarrolla la propuesta de una aplicación móvil multiplataforma orientada a mejorar la adherencia a los tratamientos médicos mediante recordatorios de medicamentos y seguimiento de pacientes. El sistema responde a la problemática frecuente de olvidos en la toma de dosis, lo que impacta negativamente en la salud de los pacientes y genera mayores costos en el sistema sanitario. La aplicación integra funcionalidades como notificaciones en tiempo real, gestión de perfiles, historial de tratamientos, reportes exportables y sincronización offline. Además, cumple con criterios de accesibilidad, seguridad y protección de datos personales, alineándose con el Objetivo de Desarrollo Sostenible N.º 3 (Salud y Bienestar). El trabajo presenta el análisis de requerimientos, modelos iniciales, diseño y planificación de desarrollo bajo la metodología ágil SCRUM, concluyendo con recomendaciones para futuras mejoras.

## Introducción

La adherencia a los tratamientos médicos constituye un reto importante en la salud pública, especialmente en pacientes crónicos y adultos mayores, quienes suelen olvidar la toma de sus medicamentos. Esta situación no solo disminuye la efectividad de los tratamientos, sino que también incrementa las complicaciones, hospitalizaciones y costos para las instituciones de salud. Frente a este problema, surge la necesidad de implementar soluciones tecnológicas que faciliten el control y el seguimiento de los tratamientos.

En este contexto, el presente informe expone el desarrollo de una aplicación móvil de recordatorio de medicamentos y seguimiento de pacientes. El proyecto busca garantizar un sistema accesible, seguro y confiable que mejore la comunicación entre médicos y pacientes, promueva hábitos de cuidado personal y contribuya al cumplimiento del ODS N.º 3 (Salud y Bienestar). Para ello, se detalla el análisis de necesidades, los requerimientos funcionales y no funcionales, los modelos iniciales del sistema, los diseños propuestos y la planificación del trabajo mediante la metodología SCRUM, asegurando un proceso de desarrollo ágil, estructurado y centrado en el usuario. (FALTA LOS DEMAS CAPITULOS DE LA INTRODUCCIÓN)

# Capítulo 1. Presentación del Proyecto

## 1.1 ODS vinculado:

La ODS elegida para el proyecto es la número 3 salud y bienestar que consiste en garantizar una vida sana y promover el bienestar para todos en todas las edades mediante el acceso universal a servicios de salud de calidad, la prevención y tratamiento de enfermedades, la reducción de la mortalidad infantil y materna, y la promoción de estilos de vida saludables.

## 1.2 Organización o institución beneficiaria:

**1.2.1 Hospitales:** Instituciones de salud que atienden a pacientes con enfermedades complejas y requieren hospitalización. Mejoran la adherencia a tratamientos y reducen reingresos.

**1.2.2 ESSALUD:** Seguro Social de Salud del Perú que brinda atención médica a trabajadores y sus familias. Facilita el monitoreo y seguimiento de pacientes en tratamientos prolongados

**1.2.3 Cruz Roja:** Organización humanitaria que realiza campañas de salud y apoyo en emergencias. Utiliza el sistema para mejorar la prevención y el seguimiento en comunidades vulnerables.

**1.2.4 Centros de salud:** Establecimientos de atención primaria que ofrecen servicios básicos de salud en zonas urbanas y rurales. Apoyan a médicos y pacientes en el control de tratamientos.

## 1.3 Problema identificado:

Los pacientes olvidan la hora exacta en la que deben tomar sus medicamentos, lo que reduce la efectividad del tratamiento. Los médicos no tienen un sistema para monitorear a sus pacientes, generando falta de contacto y retroalimentación.

## 1.4 Solución propuesta EXPLICADO A DETALLE:

La solución propuesta es una aplicación móvil multiplataforma que permite registrar medicamentos, dosis y horarios, enviar notificaciones y alarmas inteligentes incluso en modo silencioso, generar historial de cumplimiento y reportes automáticos, y funcionar sin conexión a internet, sincronizando los datos automáticamente al recuperar la conexión.

## Capítulo 2. Análisis de Necesidades y Requerimientos

### 2.1 Descripción del problema:

El problema principal es que muchos adultos mayores y pacientes crónicos suelen olvidar la toma de sus medicamentos, lo que genera baja adherencia a los tratamientos, recaídas y resistencia a los medicamentos. Esta situación incrementa las hospitalizaciones y los costos en el sistema de salud pública, afectando tanto a los pacientes como a las instituciones de salud.

### 2.2 Necesidades de los usuarios:

- ❖ Recordatorios confiables y persistentes, incluso si el dispositivo está en modo silencioso.
- ❖ Interfaz intuitiva, accesible y adaptada a personas con discapacidad visual o motriz.
- ❖ Acceso a historial y reportes para consulta médica.
- ❖ Bajo consumo de batería y funcionamiento sin conexión a internet.

### 2.3 Requerimientos funcionales (RF):

- ❖ RF1: El sistema debe permitir al usuario registrar medicamentos, especificando nombre genérico, dosis, frecuencia, vía de administración y duración del tratamiento.
- ❖ RF2: El sistema debe permitir enviar notificaciones y alarmas en tiempo real para recordar la toma de cada medicamento, incluso cuando el dispositivo se encuentre en modo silencioso.
- ❖ RF3: El sistema debe mostrar un historial de los medicamentos tomados y olvidados, y permitir exportar reportes en PDF o Excel.
- ❖ RF4: El sistema debe permitir la configuración de esquemas posológicos complejos, incluyendo excepciones, reprogramación temporal de dosis y función de posponer con límites definidos.
- ❖ RF5: El sistema debe permitir exportar y compartir de forma segura el historial de adherencia y efectos adversos en formatos estándar, con consentimiento explícito del paciente y registro en el historial de auditoría.

**Must have:**

- RF1: Registrar medicamentos (nombre genérico, dosis, frecuencia, vía, duración).
- RF2: Enviar notificaciones y alarmas en tiempo real para recordar la toma (incluso en modo silencioso).
- RF3: Mostrar historial de medicamentos tomados y olvidados.

**Should have:**

- RF4: Configuración de esquemas posológicos complejos (excepciones, reprogramaciones, posponer dosis).
- RF5: Exportar y compartir historial de adherencia en formatos estándar con consentimiento del paciente.

**Could have:**

- Exportar reportes en PDF o Excel desde el historial.

**2.4 Requerimientos no funcionales (RNF):**

- ❖ RNF1: El sistema debe ser intuitivo y accesible para personas de todas las edades, asegurando una interfaz amigable y cumpliendo con las pautas de accesibilidad para personas con discapacidad visual o motriz.
- ❖ RNF2: El sistema debe responder a las acciones del usuario en menos de 2 segundos y consumir menos del 5% de la batería diaria del dispositivo.
- ❖ RNF3: El sistema debe cifrar toda la información médica y personal en tránsito y en reposo mediante protocolos seguros y gestión periódica de claves.
- ❖ RNF4: El sistema debe garantizar una disponibilidad mínima del 99 por ciento mensual en notificaciones y asegurar su entrega en un máximo de 60 segundos desde el horario programado.

- ❖ RNF5: El sistema debe cumplir criterios de accesibilidad nivel AA, soportar localización en español de Perú y permitir operación básica sin conexión con sincronización automática al restablecer la conectividad.

**Must have:**

- RNF1: Sistema intuitivo y accesible para todas las edades (cumple pautas de accesibilidad).
- RNF2: Cifrado de toda la información médica y personal en tránsito y reposo.
- RNF3: Disponibilidad mínima del 99% en notificaciones y entrega en máximo 60 segundos.

**Should have:**

- RNF4: Respuesta en menos de 2 segundos y consumo <5% de batería.
- RNF5: Cumplir criterios de accesibilidad nivel AA y soportar localización en español de Perú.

**Could have:**

- Operación básica sin conexión y sincronización automática al restablecer conectividad.

## **2.5 Requerimientos de dominio:**

- ❖ RD1: El sistema debe cumplir con la Ley de Protección de Datos Personales del Perú (Ley N.º 29733) garantizando la privacidad de la información médica y personal del paciente.
- ❖ RD2: El sistema debe garantizar la unicidad del registro de cada paciente, impidiendo la creación de historiales médicos duplicados y asegurando la trazabilidad de todos los tratamientos asociados a un único identificador.
- ❖ RD3: El sistema debe permitir la configuración precisa de esquemas posológicos, incluyendo dosis, frecuencia, vía de administración, horario y duración, de acuerdo con las indicaciones médicas establecidas.



- ❖ RD4: El sistema debe registrar de forma inalterable todas las acciones realizadas por el paciente respecto a la toma de medicamentos, incluyendo confirmaciones, omisiones o reprogramaciones, con sello de tiempo verificable.
- ❖ RD5: El sistema debe emitir notificaciones de recordatorio en los momentos exactos definidos por el tratamiento, asegurando su ejecución incluso en condiciones de ahorro de energía o configuraciones de silencio del dispositivo, siempre que el usuario lo autorice.

**Must have:**

- RD1: Cumplimiento con la Ley de Protección de Datos Personales del Perú (Ley N.º 29733).
- RD2: Garantizar unicidad de registro de cada paciente (evitar duplicados).
- RD3: Registrar de forma inalterable todas las acciones del paciente con sello de tiempo verificable.
- RD4: Emitir notificaciones en los momentos exactos definidos por el tratamiento.

**Should have:**

- RD5: Configuración precisa de esquemas posológicos según indicaciones médicas.

**Could have:**

- (Ninguno en esta fase, porque los requisitos de dominio son casi todos obligatorios por ley).

<b>Requisito</b>	<b>Criterio de Aceptación</b>	<b>Prioridad (MoSCoW)</b>
Registrar medicamentos (nombre, dosis, frecuencia, vía y duración).	El sistema guarda correctamente todos los campos del medicamento y no permite valores nulos.	Must Have
Enviar notificaciones y alarmas en tiempo real (modo silencioso incluido).	El recordatorio suena/vibra en el horario programado incluso en modo silencioso.	Must Have
Mostrar historial de medicamentos tomados/olvidados y exportar reportes.	Se visualiza lista de tomas y omisiones, con opción de exportar a PDF/Excel.	Must Have
Configuración de esquemas posológicos complejos.	El sistema permite configurar excepciones, posponer dosis y reprogramar horarios.	Should Have
Exportar y compartir historial de adherencia de forma segura.	Solo se comparte con consentimiento, con registro de auditoría.	Must Have
Intuitivo y accesible para todas las edades y discapacidad.	La navegación es clara, con botones grandes y accesibles.	Must Have
Respuesta menor a 2 segundos y bajo consumo de batería.	El sistema responde a cualquier acción en <2s y muestra reporte de uso energético.	Must Have
Cifrado de información médica en tránsito y en reposo.	Toda comunicación usa HTTPS/TLS y base de datos con cifrado AES-256.	Must Have

Disponibilidad 99% y notificaciones ≤60s.	Reportes de uptime ≥99% y pruebas de entrega en <60s.	Must Have
Cumplir accesibilidad nivel AA, localización español Perú y operación offline.	El sistema soporta idioma español (Perú), accesibilidad AA y funciones offline básicas.	Must Have
Cumplir Ley de Protección de Datos Personales (Ley 29733).	Se valida consentimiento informado y almacenamiento con políticas de privacidad activas.	Must Have
Garantizar unicidad de registro del paciente.	No se permiten duplicados de identificadores únicos de pacientes.	Must Have
Configuración precisa de esquemas posológicos médicos.	Los campos registrados coinciden con indicaciones médicas (dosis, frecuencia, vía, horario, duración).	Must Have
Registro inalterable de acciones del paciente (confirmaciones, omisiones).	Toda acción queda registrada con sello de tiempo y no puede ser editada.	Must Have
Notificaciones exactas en horarios definidos.	El recordatorio llega en la hora exacta incluso en ahorro de energía o silencio.	Must Have

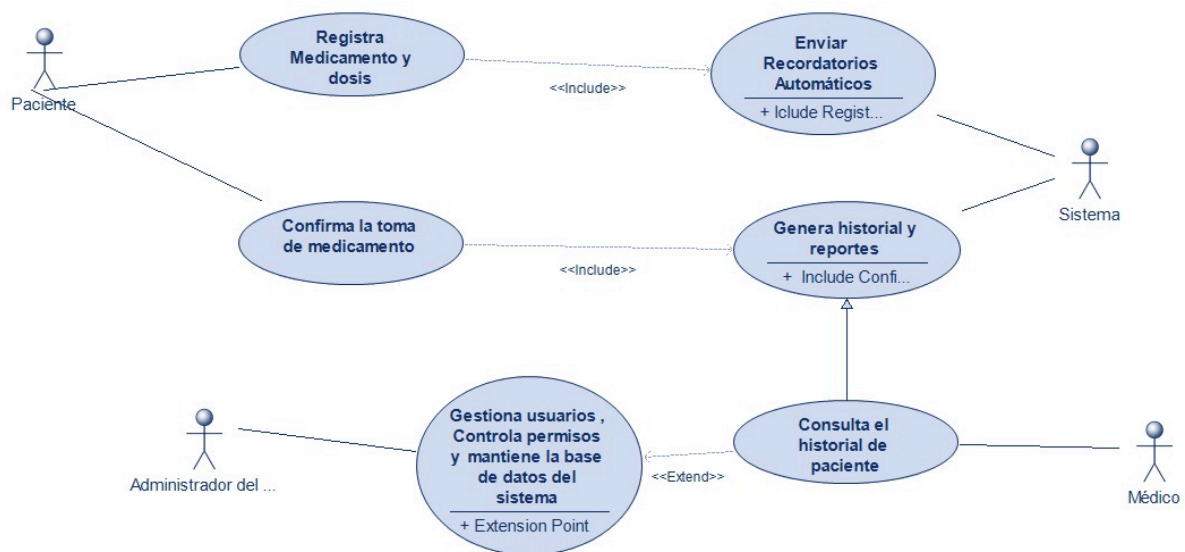
## Capítulo 3. Modelos Iniciales del Sistema

### 3.1 Modelo funcional (Diagrama de contexto y casos de uso generales)

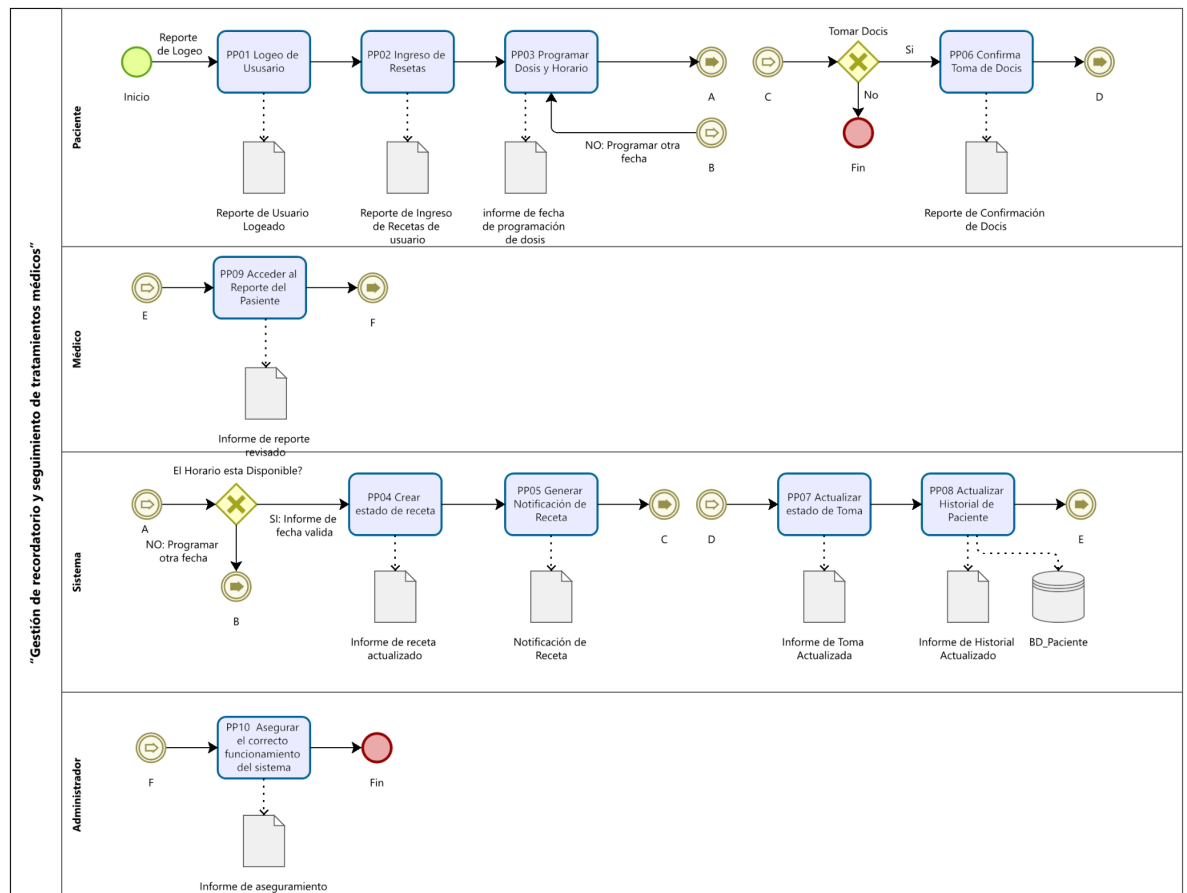
#### Actores:

- **Paciente:** Registra su tratamiento, recibe notificaciones y confirma tomas de medicamentos.
- **Médico:** Supervisa el historial de los pacientes y genera reportes de seguimiento.
- **Sistema:** Gestiona notificaciones, almacena información y genera reportes automáticos.
- **Administrador del programa:** Gestiona usuarios, controla permisos y mantiene la base de datos del sistema.

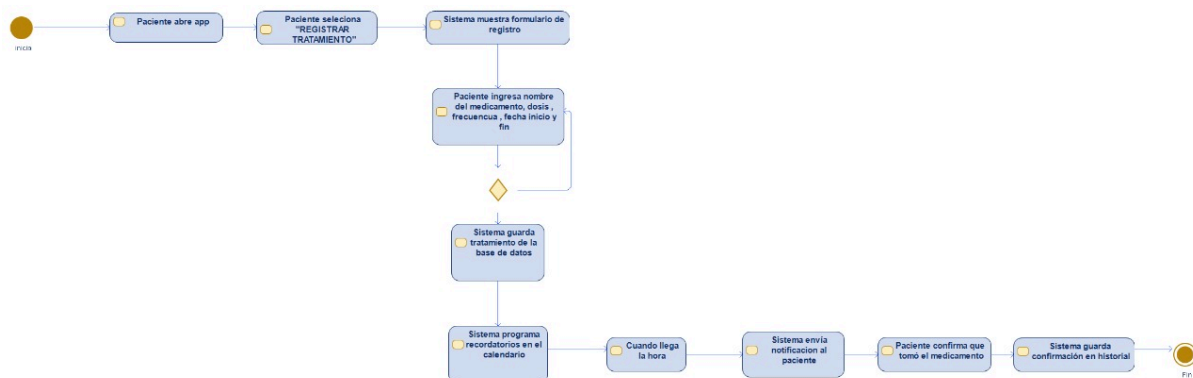
#### Casos de uso generales (CSOO1):



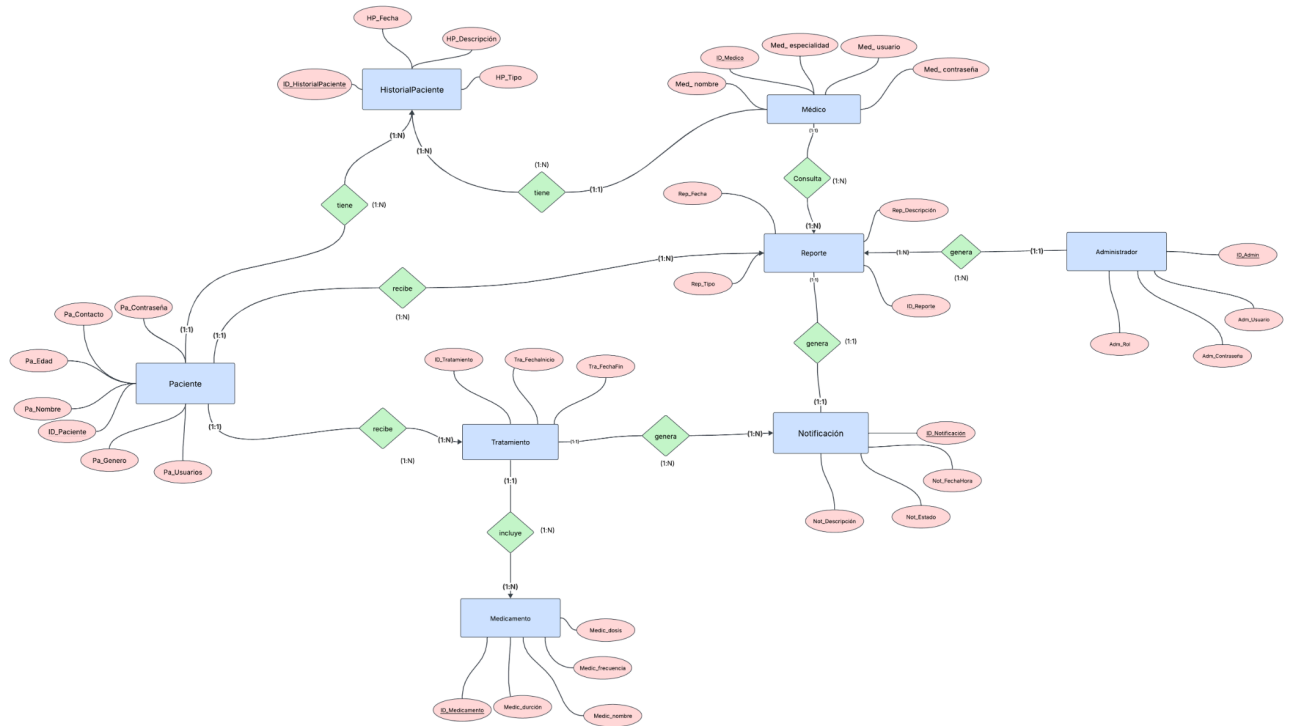
## 3.2 Modelo de procesos



## 3.3 Diagrama de actividad UML

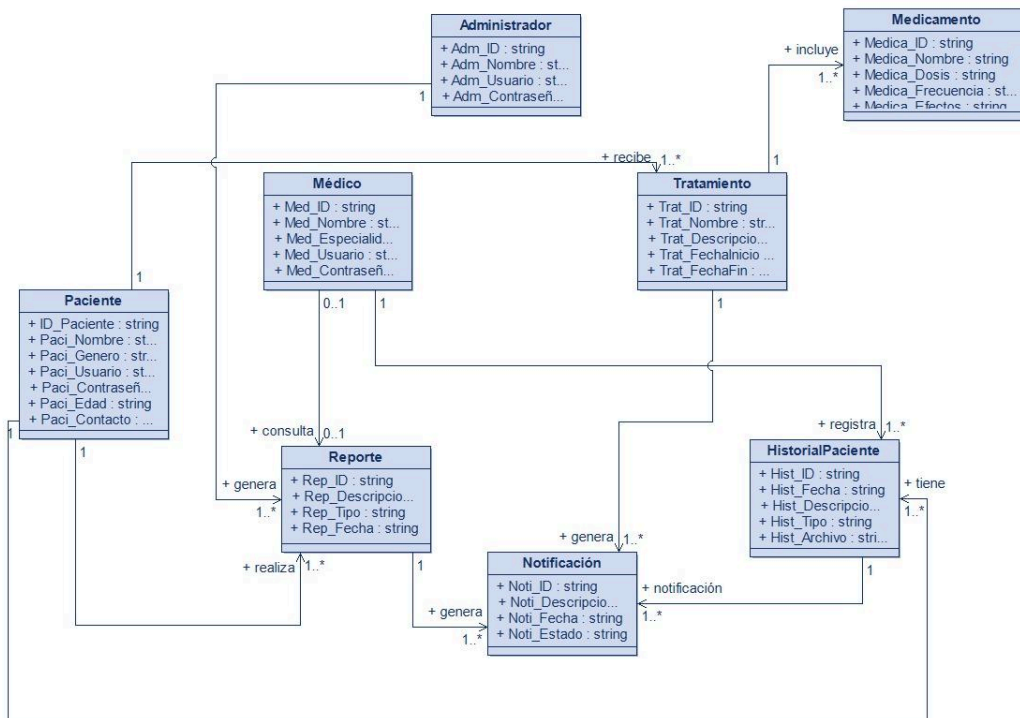


### 3.4 Modelo de datos (Modelo E-R)

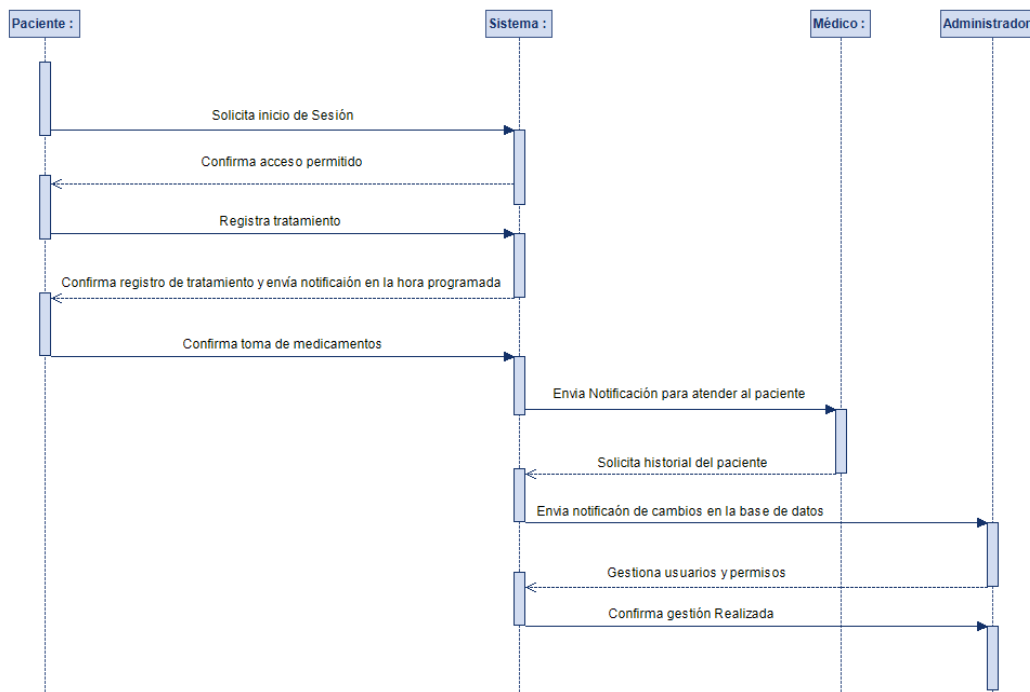


## Capítulo 4. Modelos de Diseño

- Modelo estructural (diagrama de clases inicial):



- Modelo de interacción (diagrama de secuencia):



## Capítulo 5. Metodología de Trabajo (SCRUM)

- **Definición de la metodología ágil usada:**  
SCRUM permite organizar el desarrollo en sprints cortos, con revisiones y mejoras constantes.
- **Backlog del producto (épicas e historias de usuario):**
  - **Épica 1: Gestión de tratamientos**
    - HU01 – Registrar medicamentos
    - HU02 – Editar o eliminar medicamentos
    - HU03 – Visualizar historial de tratamientos
  - **Épica 2: Recordatorios automáticos**
    - HU04 – Notificaciones de toma de medicamentos
    - HU05 – Configurar frecuencia y duración de recordatorios
    - HU06 – Alertas a familiares por dosis omitida
  - **Épica 3: Supervisión médica**
    - HU07 – Visualizar progreso del paciente
    - HU08 – Enviar recomendaciones al paciente
    - HU09 – Generar reportes de adherencia
  - **Épica 4: Gestión de usuarios y seguridad**
    - HU10 – Registro de usuario
    - HU11 – Inicio de sesión seguro
    - HU12 – Recuperación de contraseña



- **Planificación de sprints:**

- **Sprint 1 (01/09/25 – 15/09/25)**

**Meta:** Implementar la base del sistema con registro y notificaciones iniciales.

- HU01 – Registrar medicamentos
- HU04 – Notificaciones de toma de medicamentos
- HU10 – Registro de usuario

- **Sprint 2 (16/09/25 – 30/09/25)**

**Meta:** Consolidar seguridad y edición de tratamientos.

- HU02 – Editar o eliminar medicamentos
- HU11 – Inicio de sesión seguro
- HU05 – Configurar frecuencia y duración de recordatorios

- **Sprint 3 (01/10/25 – 15/10/25)**

**Meta:** Mejorar la gestión con historial, alertas y progreso médico.

- HU03 – Visualizar historial de tratamientos
- HU06 – Alertas a familiares por dosis omitida
- HU07 – Visualizar progreso del paciente

- **Sprint 4 (16/10/25 – 31/10/25)**

**Meta:** Completar supervisión avanzada y seguridad de acceso.

- HU08 – Enviar recomendaciones al paciente
- HU09 – Generar reportes de adherencia
- HU12 – Recuperación de contraseña

# Capítulo 6. Diseño de Arquitectura y Patrones

## 6.1 Estrategia de diseño del software

Las **estrategias de diseño** son métodos que orientan la forma en que se aborda la estructuración de un sistema. Permiten organizar la solución de acuerdo con la complejidad del problema, el tipo de software y los objetivos de desarrollo. En este proyecto se consideraron las siguientes estrategias:

- **Estrategia Ascendente (Bottom-Up)**

Parte del **diseño de los componentes más básicos o reutilizables**, que luego se integran en estructuras más complejas.

Se usó para desarrollar los módulos individuales los cuales luego fueron integrados en la capa de lógica de negocio.

- **Estrategia Orientada a Objetos**

Organiza el sistema en **objetos que combinan datos y comportamientos**, favoreciendo la reutilización, el encapsulamiento y la extensibilidad del código.

Se utilizó para modelar las clases principales del sistema: Usuario, Medicamento, Tratamiento y Notificación.

### Conclusión:

El sistema aplica principalmente una **estrategia descendente y orientada a objetos**, ya que se inició con el diseño general del sistema (arquitectura por capas) y luego se desarrollaron los objetos y clases específicos que implementan la funcionalidad completa.

## 6.2 Tipo de arquitectura de sistema:

El sistema se diseñó bajo una **arquitectura en tres capas (3-tier)**, lo cual permite la separación clara entre la interfaz de usuario, la lógica de negocio y el acceso a datos. Este enfoque mejora la mantenibilidad, escalabilidad y seguridad del sistema, garantizando un desarrollo modular y flexible ante futuras mejoras.

El sistema está compuesto por:

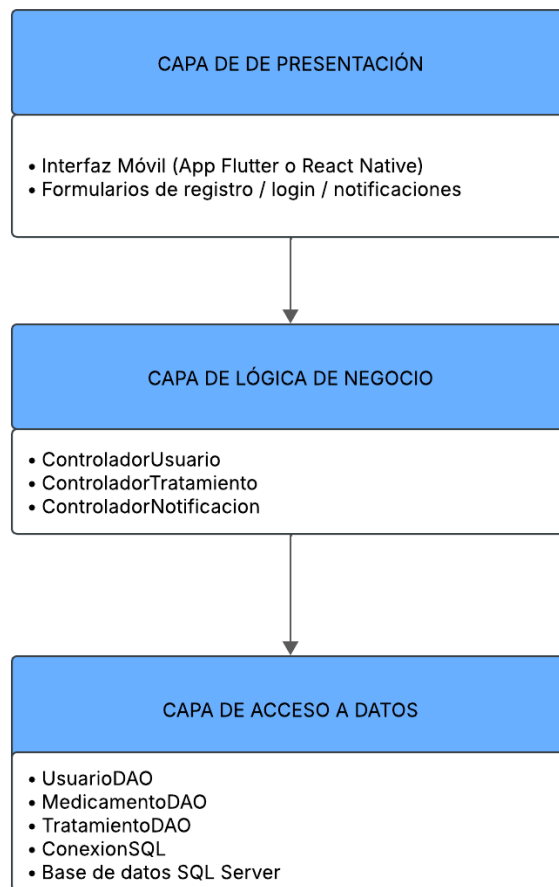
- **Capa de presentación:** encargada de la interacción con el usuario a través de una interfaz móvil multiplataforma. Gestiona la visualización de los recordatorios, reportes y notificaciones.
- **Capa de lógica de negocio:** contiene las reglas y procesos que validan, calculan y controlan la adherencia a los tratamientos. Aquí se ubican los controladores de tratamiento, validadores de dosis y servicios de notificación.

- **Capa de acceso a datos:** maneja la conexión con la base de datos en **SQL Server**, encargándose de ejecutar consultas, procedimientos almacenados y transacciones seguras mediante la clase Conexión SQL.

Este modelo facilita la reutilización de componentes y permite implementar pruebas unitarias de manera independiente en cada capa.

#### Justificación de la arquitectura por capas

Criterio	Arquitectura por Capas
<b>Separación lógica</b>	Alta, las tres capas son independientes y desacopladas.
<b>Escalabilidad</b>	Facilita la creación de servicios adicionales (API, módulos web).
<b>Integración con SQL Server</b>	Directa mediante la capa de datos.
<b>Reutilización</b>	Alta, los módulos pueden ser compartidos con otros sistemas.
<b>Seguridad</b>	Mayor, se controlan los accesos en la capa de lógica.

**Conclusión:**

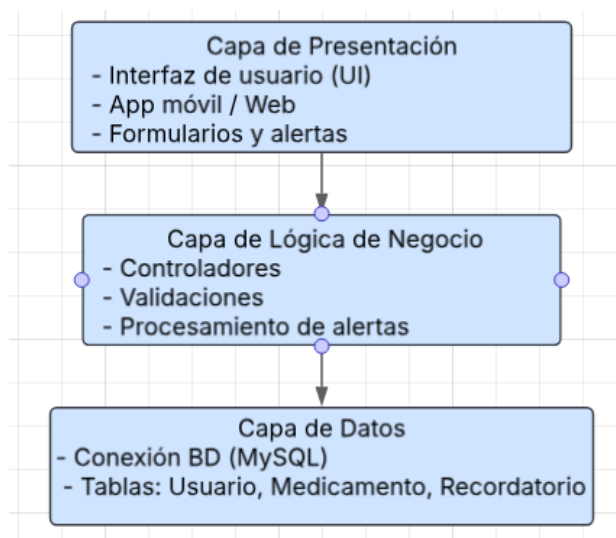
La arquitectura por capas se adapta mejor al contexto del proyecto, que requiere conexión directa a SQL Server, soporte offline y sincronización, manteniendo una estructura limpia y mantenible.

Patrón	Descripción	Aplicación en el sistema
Capas	Divide el sistema en capas independientes (presentación, lógica de negocio, datos), facilitando el mantenimiento y la escalabilidad.	Se aplica en toda la plataforma: la capa de presentación muestra la interfaz web/móvil; la capa de lógica gestiona recordatorios, alertas y usuarios; y la capa de datos maneja el acceso a la base de datos MySQL/Firebase.

### 6.3 Patrones de diseño aplicados

Para optimizar el diseño y la modularidad, se aplicaron los siguientes **patrones de diseño**:

#### -Diagrama del Patrón en Capas



### 6.4 Diseño estructural

**Descripción textual del diagrama:**

#### - Clase Administrador

**Atributos:**

- idAdministrador
- nombre
- usuario
- contraseña

**Relación principal:**

- Un administrador genera múltiples reportes.

## **- Clase Médico**

### **Atributos:**

- idMedico
- nombre
- especialidad
- correo
- usuario

### **Relación principal:**

- Un médico atiende a varios pacientes.
- Un médico crea múltiples reportes.

## **- Clase Paciente**

### **Atributos:**

- idPaciente
- nombre
- correo
- contraseña

### **Relación principal:**

- Un paciente recibe varios tratamientos.
- Un paciente genera un historial clínico.
- Un paciente recibe uno o muchos reportes

## **- Clase Medicamento**

### **Atributos:**

- idMedicamento
- nombre
- dosis
- frecuencia
- duracion
- viaAdministracion

### **Relación principal:**

- Un medicamento puede pertenecer a varios tratamientos.

## **- Clase Tratamiento**

### **Atributos:**

- idTratamiento
- idPaciente
- idMedico
- fechaInicio
- horaRecordatorio
- estadoDosis

**Relación principal:**

- Un tratamiento incluye varios medicamentos.
- Un paciente puede tener varios tratamientos.
- Un médico puede asignar muchos tratamientos.

**- Clase Notificacion**

**Atributos:**

- idNotificacion
- mensaje
- fechaHora
- idPaciente

**Relación principal:**

- Un paciente recibe múltiples notificaciones.
- Cada tratamiento puede generar notificaciones.

**- Clase HistorialPaciente**

**Atributos:**

- idHistorial
- idPaciente
- fecha
- descripcion

**Relación principal:**

- Un paciente tiene un historial con muchos registros.
- Los tratamientos pueden agregar entradas al historial.

**- Clase Reporte**

**Atributos:**

- idReporte
- idMedico
- fecha
- diagnostico

## Relación principal:

- Un médico genera varios reportes.
- Un reporte puede estar asociado a un paciente o a un tratamiento.

## 6.5 diagramas de componentes, despliegue o infraestructura

# CAPÍTULO 7. DISEÑO DETALLADO DE LA BASE DE DATOS

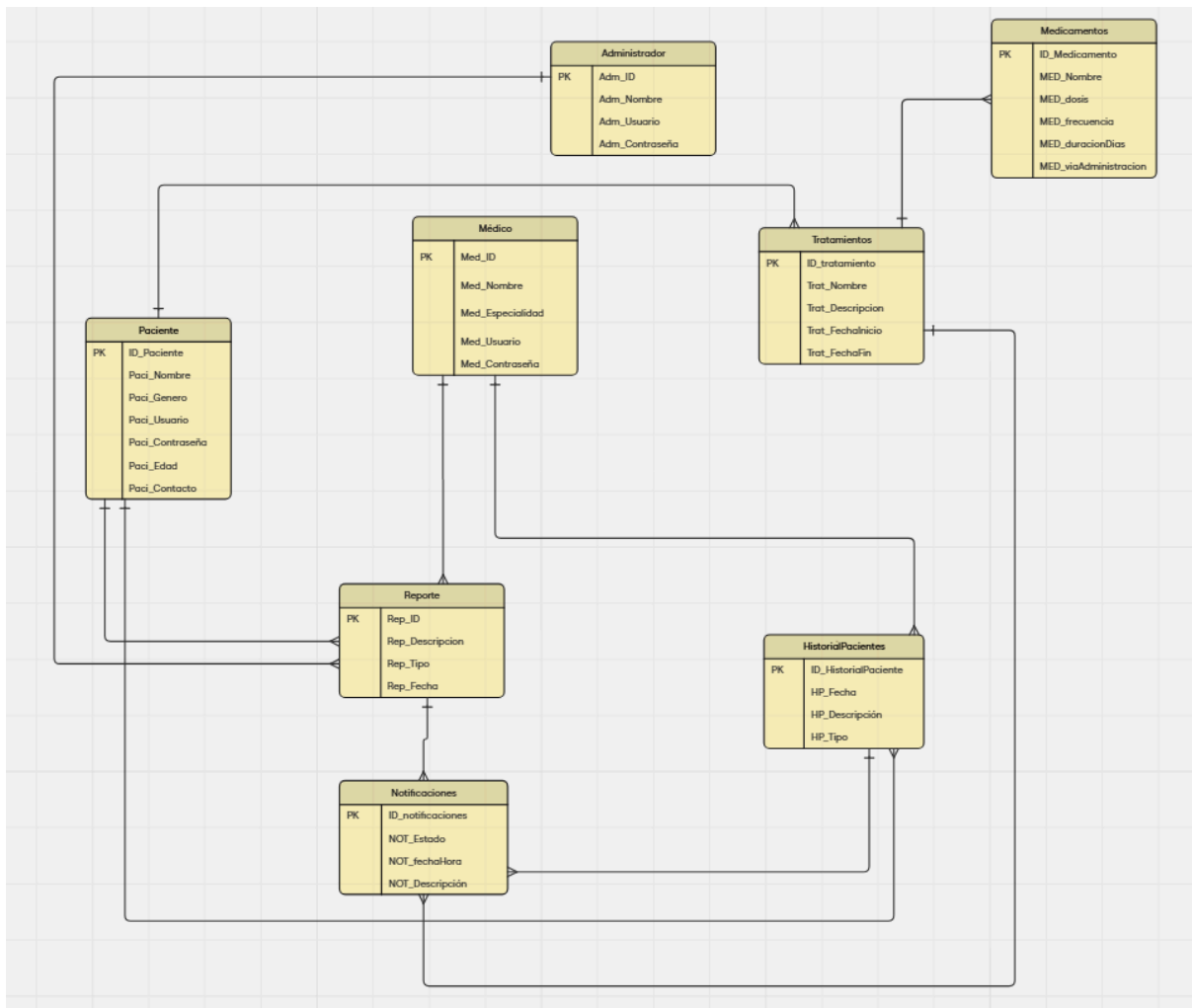
Link de normalización:

1 Normalización																															
Admin_ID	Admin_Nom	Admin_Usua	Admin_Cor_Pasien	Med_ID	Med_Nom	Med_Espe	Med_Usua	Med_Cor_Pasien	ID_Pasien	Paci_Nom	Paci_Sexo	Paci_Usua	Paci_Cor_Pasien	Paci_Edad	Trat_ID	Trat_Nom	Trat_Descri	Trat_Fecha	Trat_Fecha	Medica_ID	Medica_Nom	Medica_Dos	Medica_Fr	Medica_Ef	Medic_ID	Medic_Nom	Medic_Dos	Medic_Fr	Medic_Ef	Resp_ID	
ADM03	Carlos Pérez	carlos12	jess123	MED02	Soía Vides	Pediatría	ivareaz	ped2023	PAC004	Diego Soto	Masculino	diego	d123	30	867854321	TR04	Control	Antihistami	14/2/2024	30-4-2024	MED07	Isuproleno	400mg	Cada 8h	Náuseas	H004	12/5/2024	Realizado de análisis	Laboratorio	analisis.pdf	H03
ADM01	Ana Torres	admin01a	qwerty	MED04	Andrés Ramos	Dermatología	arancos	derma55	PAC002	María López	Femenino	mlopez	mar-24	22	912345678	TR02	Control	Antipruriti	10/3/2024	10/3/2024	MED03	Loratadina	10mg	Cada 24h	Somnolencia	H001	8/1/2024	Examen de piel	Consulta	inicio.pdf	R01
ADM04	Luis Gómez	luisg	12345	MED01	José Medina	Cardiología	medina	cardio123	PAC001	Pedro Torres	Masculino	pedro_t	p88	27	887111222	TR01	Control	Presión	5/1/2024	5/5/2024	MED01	Enalapril	5mg	Cada 12h	Mareos	H003	15/3/2024	Evolución clínica	Evaluación	notas3.doc	R04
ADM02	Marta Ruiz	mruiz	admin2024	MED03	Laura Silva	Neurología	neu	neuro78	PAC003	Ana Castillo	Femenino	acas	ana55	24	945789123	TR03	Evaluación	Pruebas cognitivas	1/2/2024	1/2/2024	MED05	Omega 3	1 cápsula	Diario	Sin efectos	H002	20/2/2024	Forma cognitiva	Pruebas	cog.pdf	R02
2 Normalización																															
Administrador				Médico				Paciente							Tratamiento					Medicamento										Histori	
Admin_ID	Admin_Nom	Admin_Usua	Admin_Cor_Pasien	Med_ID	Med_Nom	Med_Espe	Med_Usua	Med_Cor_Pasien	ID_Pasien	Paci_Nom	Paci_Sexo	Paci_Usua	Paci_Cor_Pasien	Paci_Edad	Trat_ID	Trat_Nom	Trat_Descri	Trat_Fecha	Trat_Fecha	Medica_ID	Medica_Nom	Medica_Dos	Medica_Fr	Medica_Ef	Medic_ID	Medic_Nom	Medic_Dos	Medic_Fr	Medic_Ef	Resp_ID	
ADM03	Carlos Pérez	carlos12	jess123	MED02	Soía Vides	Pediatría	ivareaz	ped2023	PAC004	Diego Soto	Masculino	diego	d123	30	867854321	TR04	Control	Antihistami	14/2/2024	30-4-2024	MED07	Isuproleno	400mg	Cada 8h	Náuseas	H004	12/5/2024	Realizado de análisis	Laboratorio	analisis.pdf	H03
ADM01	Ana Torres	admin01a	qwerty	MED04	Andrés Ramos	Dermatología	arancos	derma55	PAC002	María López	Femenino	mlopez	mar-24	22	912345678	TR02	Control	Antipruriti	10/3/2024	10/3/2024	MED03	Loratadina	10mg	Cada 24h	Somnolencia	H001	8/1/2024	Examen de piel	Consulta	inicio.pdf	R01
ADM04	Luis Gómez	luisg	12345	MED01	José Medina	Cardiología	medina	cardio123	PAC001	Pedro Torres	Masculino	pedro_t	p88	27	887111222	TR01	Control	Presión	5/1/2024	5/5/2024	MED01	Enalapril	5mg	Cada 12h	Mareos	H003	15/3/2024	Evolución clínica	Evaluación	notas3.doc	R04
ADM02	Marta Ruiz	mruiz	admin2024	MED03	Laura Silva	Neurología	neu	neuro78	PAC003	Ana Castillo	Femenino	acas	ana55	24	945789123	TR03	Evaluación	Pruebas cognitivas	1/2/2024	1/2/2024	NULL					H002					
3 Normalización																															
Administrador				Médico				Paciente							Tratamiento					Medicamento										Histori	
Admin_ID	Admin_Nom	Admin_Usua	Admin_Cor_Pasien	Med_ID	Med_Nom	Med_Espe	Med_Usua	Med_Cor_Pasien	ID_Pasien	Paci_Nom	Paci_Sexo	Paci_Usua	Paci_Cor_Pasien	Paci_Edad	Trat_ID	Trat_Nom	Trat_Descri	Trat_Fecha	Trat_Fecha	Medica_ID	Medica_Nom	Medica_Dos	Medica_Fr	Medica_Ef	Medic_ID	Medic_Nom	Medic_Dos	Medic_Fr	Medic_Ef	Resp_ID	
ADM03	Carlos Pérez	carlos12	jess123	MED02	Soía Vides	Pediatría	ivareaz	ped2023	PAC004	Diego Soto	Masculino	diego	d123	30	867854321	TR04	Control	Antihistami	14/2/2024	30-4-2024	MED07	Isuproleno	400mg	Cada 8h	Náuseas	H004	12/5/2024	Realizado de análisis	Laboratorio	analisis.pdf	H03
ADM01	Ana Torres	admin01a	qwerty	MED04	Andrés Ramos	Dermatología	arancos	derma55	PAC002	María López	Femenino	mlopez	mar-24	22	912345678	TR02	Control	Antipruriti	10/3/2024	10/3/2024	MED03	Loratadina	10mg	Cada 24h	Somnolencia	H001	8/1/2024	Examen de piel	Consulta	inicio.pdf	R01
ADM04	Luis Gómez	luisg	12345	MED01	José Medina	Cardiología	medina	cardio123	PAC001	Pedro Torres	Masculino	pedro_t	p88	27	887111222	TR01	Control	Presión	5/1/2024	5/5/2024	MED01	Enalapril	5mg	Cada 12h	Mareos	H003	15/3/2024	Evolución clínica	Evaluación	notas3.doc	R04
ADM02	Marta Ruiz	mruiz	admin2024	MED03	Laura Silva	Neurología	neu	neuro78	PAC003	Ana Castillo	Femenino	acas	ana55	24	945789123	TR03	Evaluación	Pruebas cognitivas	1/2/2024	1/2/2024	NULL					H002					

[https://docs.google.com/spreadsheets/d/1puflgQP0zGxzzJtWO38Ys6eSrudY9DVc/edit?usp=drive link&ouid=116649432003947337904&rtpof=true&sd=true](https://docs.google.com/spreadsheets/d/1puflgQP0zGxzzJtWO38Ys6eSrudY9DVc/edit?usp=drive_link&ouid=116649432003947337904&rtpof=true&sd=true)



## 7.1 Modelo lógico



El modelo lógico se compone de las siguientes tablas principales:

### 1. **Administrador**

- Adm\_ID
- Adm\_Nombre
- Adm\_Usuario
- Adm\_Contrasena

### 2. **Médico**

- Med\_ID
- Med\_Nombre

- Med\_Especialidad
- Med\_Usuario
- Med\_Contrasena

### **3. Paciente**

- ID\_Paciente
- Paci\_Nombre
- Paci\_Genero
- Paci\_Usuario
- Paci\_Contrasena
- Paci\_Edad
- Paci\_Contacto

### **4. Tratamiento**

- Trat\_ID
- Trat\_Nombre
- Trat\_Descripcion
- Trat\_FechaInicio
- Trat\_FechaFin
- Med\_ID (*relación con Médico*)

### **5. Medicamento**

- Medica\_ID
- Medica\_Nombre
- Medica\_Dosis
- Medica\_Frecuencia
- Medica\_Efectos
- Trat\_ID (*relación con Tratamiento*)

## 6. HistorialPaciente

- *Hist\_ID*
- *Hist\_Fecha*
- *Hist\_Descripcion*
- *Hist\_Tipo*
- *ID\_Paciente* (*relación con Paciente*)
- *Trat\_ID* (*relación con Tratamiento*)

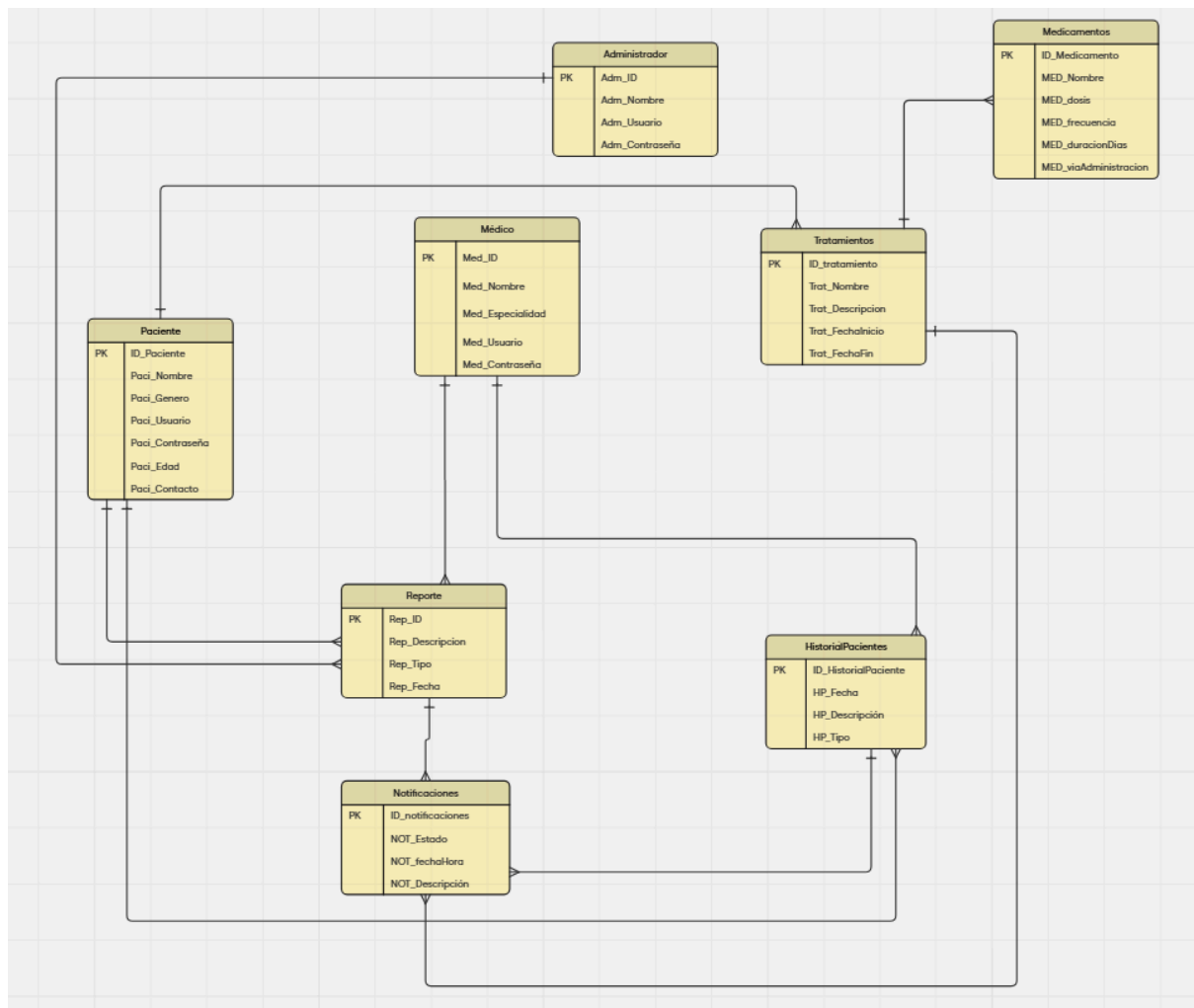
## 7. Reporte

- *Rep\_ID*
- *Rep\_Descripcion*
- *Rep\_Tipo*
- *Rep\_Fecha*
- *Med\_ID* (*relación con Médico*)
- *ID\_Paciente* (*relación con Paciente*)

## 8. Notificacion

- *Noti\_ID*
- *Noti\_Descripcion*
- *Noti\_Fecha*
- *Noti\_Estado*
- *ID\_Paciente* (*relación con Paciente*)
- *Med\_ID* (*relación con Médico*)

## 7.2 Modelo físico (fragmento SQL Server)



```

CREATE TABLE Administrador (
    Adm_ID VARCHAR(20) PRIMARY KEY,
    Adm_Nombre VARCHAR(80) NOT NULL,
    Adm_Usuario VARCHAR(50) NOT NULL,
    Adm_Contrasena VARCHAR(100) NOT NULL
);
GO

```

```

CREATE TABLE Medico (
    Med_ID VARCHAR(20) PRIMARY KEY,
    Med_Nombre VARCHAR(80) NOT NULL,
    Med_Especialidad VARCHAR(80) NOT NULL,

```

```
Med_Usuario VARCHAR(50) NOT NULL,  
Med_Contrasena VARCHAR(100) NOT NULL  
);  
GO
```

```
CREATE TABLE Paciente (  
    ID_Paciente VARCHAR(20) PRIMARY KEY,  
    Paci_Nombre VARCHAR(80) NOT NULL,  
    Paci_Genero VARCHAR(20) NULL,  
    Paci_Usuario VARCHAR(50) NOT NULL,  
    Paci_Contrasena VARCHAR(100) NOT NULL,  
    Paci_Edad INT NULL,  
    Paci_Contacto VARCHAR(80) NULL  
);  
GO
```

```
CREATE TABLE Tratamiento (  
    Trat_ID VARCHAR(20) PRIMARY KEY,  
    Trat_Nombre VARCHAR(80) NOT NULL,  
    Trat_Descripcion VARCHAR(200) NULL,  
    Trat_FechaInicio DATE NOT NULL,  
    Trat_FechaFin DATE NULL,  
    Med_ID VARCHAR(20) NOT NULL,  
    FOREIGN KEY (Med_ID) REFERENCES Medico(Med_ID)  
);  
GO
```

```
CREATE TABLE Medicamento (  

```

```

Medica_ID VARCHAR(20) PRIMARY KEY,
Medica_Nombre VARCHAR(80) NOT NULL,
Medica_Dosis VARCHAR(50) NULL,
Medica_Frecuencia VARCHAR(50) NULL,
Medica_Efectos VARCHAR(200) NULL,
Trat_ID VARCHAR(20) NOT NULL,
FOREIGN KEY (Trat_ID) REFERENCES Tratamiento(Trat_ID)
);
GO

```

```

CREATE TABLE HistorialPaciente (
    Hist_ID VARCHAR(20) PRIMARY KEY,
    Hist_Fecha DATE NOT NULL,
    Hist_Descripcion VARCHAR(200) NULL,
    Hist_Tipo VARCHAR(50) NOT NULL,
    ID_Paciente VARCHAR(20) NOT NULL,
    Trat_ID VARCHAR(20) NOT NULL,
    FOREIGN KEY (ID_Paciente) REFERENCES Paciente(ID_Paciente),
    FOREIGN KEY (Trat_ID) REFERENCES Tratamiento(Trat_ID)
);
GO

```

```

CREATE TABLE Reporte (
    Rep_ID VARCHAR(20) PRIMARY KEY,
    Rep_Descripcion VARCHAR(200) NULL,
    Rep_Tipo VARCHAR(50) NOT NULL,
    Rep_Fecha DATE NOT NULL,
    Med_ID VARCHAR(20) NOT NULL,

```

```

ID_Paciente VARCHAR(20) NOT NULL,
FOREIGN KEY (Med_ID) REFERENCES Medico(Med_ID),
FOREIGN KEY (ID_Paciente) REFERENCES Paciente(ID_Paciente)
);
GO

```

```

CREATE TABLE Notificacion (
    Noti_ID VARCHAR(20) PRIMARY KEY,
    Noti_Descripcion VARCHAR(200) NOT NULL,
    Noti_Fecha DATE NOT NULL,
    Noti_Estado VARCHAR(50) NOT NULL,
    ID_Paciente VARCHAR(20) NOT NULL,
    Med_ID VARCHAR(20) NULL,
    FOREIGN KEY (ID_Paciente) REFERENCES Paciente(ID_Paciente),
    FOREIGN KEY (Med_ID) REFERENCES Medico(Med_ID)
);
GO

```

#### **7.4 Procedimientos almacenados, vistas y triggers**

- **Procedimientos almacenados (sp\_ActualizarPaciente):**

```
-- 2- Actualizar datos completos de paciente
CREATE PROCEDURE sp_ActualizarPaciente
    @ID_Paciente    VARCHAR(20),
    @Nombre          VARCHAR(80),
    @Genero          VARCHAR(20),
    @Usuario         VARCHAR(50),
    @Contrasena      VARCHAR(100),
    @Edad            INT,
    @Contacto        VARCHAR(80)
AS
BEGIN
    UPDATE Paciente
    SET
        Paci_Nombre      = @Nombre,
        Paci_Genero      = @Genero,
        Paci_Usuario     = @Usuario,
        Paci_Contrasena = @Contrasena,
        Paci_Edad        = @Edad,
        Paci_Contacto    = @Contacto
    WHERE ID_Paciente = @ID_Paciente;
    PRINT 'Paciente actualizado correctamente.';
END;
```

8 %

Messages

Commands completed successfully.

Completion time: 2025-11-26T00:14:45.9199978-05:00

```
EXECUTE sp_ActualizarPaciente
    @ID_Paciente = 'PAC001',
    @Nombre = 'Carlos Mendoza',
    @Genero = 'Masculino',
    @Usuario = 'carlosm',
    @Contrasena = 'nuevaClave123',
    @Edad = 32,
    @Contacto = '987654321';
```

8 %

Messages

(1 row affected)  
Paciente actualizado correctamente.

Completion time: 2025-11-26T00:16:24.5986415-05:00

- Vistas (Vista\_Tratamientos\_Activos):



```
-- Vista 1: Tratamientos activos (aún en curso)
CREATE VIEW Vista_Tratamientos_Activos AS
SELECT T.Trat_ID, T.Trat_Nombre, T.Trat_Descripcion, T.Trat_FechaInicio, T.Trat_FechaFin, M.Med_Nombre
AS Medico_Encargado
FROM Tratamiento T
INNER JOIN Medico M ON T.Med_ID = M.Med_ID
WHERE T.Trat_FechaFin IS NULL; -- Tratamiento activo

-- Probar la vista
SELECT * FROM Vista_Tratamientos_Activos;
```

131 %

Messages

Commands completed successfully.

Completion time: 2025-11-26T00:22:04.1665173-05:00

```
-- Probar la vista
SELECT * FROM Vista_Tratamientos_Activos;
```

.31 %

Results Messages

	Trat_ID	Trat_Nombre	Trat_Descripcion	Trat_FechaInicio	Trat_FechaFin	Medico_Encargado
1	T002	Tratamiento Pediátrico	Fiebre persistente	2025-02-05	NULL	Dra. Sofía Ruiz
2	T004	Tratamiento Neurológico	Migrañas severas	2025-03-01	NULL	Dra. Ana Morales

#### - Triggers (trg\_ValidarEdadPaciente):

```
CREATE TRIGGER trg_ValidarEdadPaciente
ON Paciente
FOR INSERT
AS
BEGIN
-- Verificar si la edad es inválida
IF EXISTS (
SELECT 1
FROM inserted
WHERE Paci_Edad < 0 OR Paci_Edad > 120
)
BEGIN
RAISERROR ('La edad del paciente no es válida (debe estar entre 0 y 120 años).', 16, 1);
ROLLBACK TRANSACTION;
RETURN;
END;

PRINT 'Paciente insertado correctamente.';
END;
```

.08 %

Messages

Commands completed successfully.

Completion time: 2025-11-26T00:34:50.2607105-05:00

## 7.5 Estrategias de seguridad y respaldo

- **Cifrado de contraseñas** usando funciones hash (HASHBYTES( ' SHA2\_256 ' , contraseña)).
- **Restricciones de integridad referencial** para evitar eliminaciones en cascada no controladas.
- **Respaldos automáticos** semanales mediante plan de mantenimiento de SQL Server.
- **Roles y permisos:**
  - rolAdmin: control total sobre todas las tablas.
  - rolMedico: solo lectura en vistas de pacientes.
  - rolPaciente: inserción en su propio historial.

## 7.5 Lecciones aprendidas

Durante el diseño de la base de datos se comprendió la importancia de normalizar las tablas para evitar redundancias. También se valoró la utilidad de las vistas para generar reportes sin comprometer la seguridad de los datos sensibles.

## CAPÍTULO 8. DISEÑO DETALLADO DE SISTEMAS EN RED Y MÓVILES

### 8.1 Modelo de comunicación

El sistema opera bajo un modelo **cliente-servidor**:

- El **cliente móvil** (app) envía peticiones mediante HTTP/HTTPS a un **API REST** alojado en un servidor web.
- El servidor procesa la lógica y accede a **SQL Server** para leer o actualizar información.
- Las notificaciones se gestionan mediante un **servicio en la nube** (Firebase Cloud Messaging o Microsoft Notification Hub).

### 8.2 Diseño del sistema móvil

La aplicación móvil contará con las siguientes interfaces principales:

Pantalla	Descripción
Inicio de sesión / Registro	Permite autenticación segura y registro de usuarios.
Dashboard principal	Muestra el resumen del tratamiento, próximos recordatorios y progreso.
Gestión de medicamentos	Registro, edición y eliminación de medicamentos.
Historial de tomas	Listado cronológico con filtros por fecha y estado.
Notificaciones	Muestra alertas y recordatorios de próximas dosis.

Perfil y configuración	Ajustes de idioma, frecuencia de recordatorios y accesibilidad.
------------------------	---

### 8.3 Gestión de datos en red

- Sincronización **bidireccional** entre el dispositivo y el servidor.
- Modo **offline**: los registros locales se almacenan en caché (SQLite interno) y se sincronizan automáticamente al recuperar la conexión.
- Manejo de **JSON** para intercambio de datos liviano y estructurado.

### 8.4 Seguridad en red y móviles

- **HTTPS obligatorio** para todas las solicitudes.
- **Tokens JWT (JSON Web Token)** para autenticación persistente.
- **Validación en dos pasos (2FA)** opcional mediante correo electrónico.
- **Cifrado local AES-256** en datos sensibles almacenados en el dispositivo.
- **Permisos controlados** según rol del usuario.

### 8.5 Justificación técnica

Se eligió un entorno multiplataforma (por ejemplo, Flutter o React Native) debido a su capacidad de ejecutar en Android e iOS con un solo código fuente, manteniendo la comunicación con un backend en .NET Core o Node.js conectado a SQL Server.

Esta decisión optimiza tiempos de desarrollo, reduce costos y permite integrar fácilmente notificaciones y servicios web.

### 8.6 Lecciones aprendidas

Durante el diseño móvil se evidenció la necesidad de priorizar la usabilidad y la accesibilidad para adultos mayores. Además, se comprobó que una arquitectura cliente-servidor con sincronización offline mejora significativamente la experiencia del usuario y la confiabilidad de la aplicación.

# **CAPÍTULO 9. Diseño de Interfaz y Experiencia de Usuario (UX/UI)**

## **9.1 Perfil del usuario / usuario meta**

### **Definición del usuario final según ODS y necesidades reales**

El sistema está orientado a tres tipos principales de usuarios, alineados al ODS 3 (Salud y Bienestar):

#### **1. Pacientes (Usuario principal)**

- Adultos mayores con dificultades para recordar la toma de sus medicamentos.
- Pacientes crónicos (diabetes, hipertensión, asma).
- Personas con tratamientos prolongados o complejos.
- Necesidades:
  - Recordatorios confiables.
  - Interfaz sencilla, letras grandes.
  - Historial claro para controlar su adherencia.

#### **2. Médicos**

- Profesionales de salud que supervisan tratamientos.
- Necesidades:
  - Revisar el progreso de los pacientes.
  - Acceder a reportes claros y exportables.
  - Enviar recomendaciones rápidas.

#### **3. Familiares / Cuidadores**

- Personas que supervisan a un adulto mayor o dependiente.
- Necesidades:
  - Ver alertas de dosis omitidas.
  - Acompañar mejor el tratamiento sin estar presentes.

## Contexto de uso del sistema

- El sistema se usa **diariamente**, en situaciones como:
  - Casa, hospital o consultorio.
  - Momentos en los que el paciente recibe recordatorios.
  - Consultas médicas para mostrar el historial.
- Dispositivo principal: smartphone Android/iOS.
- Condiciones:
  - Personas con visión reducida → requiere accesibilidad.
  - Personas que no son expertas en tecnología → interfaz simple.
  - Puede usarse en zonas sin internet → funcionalidad offline.

## 9.2 Principios de diseño aplicados (HCI) y cómo se aplican en el prototipo

### 1. Consistencia

- Se mantienen **colores uniformes** para cada módulo (azul = salud, verde = confirmación).
- Botones principales siempre están en la misma ubicación.
- Todas las pantallas usan la misma tipografía y estructura.

- Íconos estandarizados: para confirmar, para editar, para eliminar.

## **2. Visibilidad**

- Las funciones esenciales (Agregar medicamento, Próxima toma, Confirmar dosis) se muestran claramente en el Dashboard.
- Tamaños grandes para:
  - Hora del recordatorio
  - Nombre del medicamento
- Notificaciones visibles incluso en modo silencioso.

## **3. Accesibilidad**

- Letras grandes, alto contraste para adultos mayores.
- Botones espaciosos para manos temblorosas.
- Soporte de lectura por voz (opcional).
- Interfaz compatible con la WCAG 2.1 nivel AA.

## **4. Control del usuario**

- El usuario puede:
  - Posponer una toma.
  - Editar medicamentos.
  - Cancelar recordatorios.
  - Exportar o borrar su historial.
- No hay procesos irreversibles sin confirmación.

## **5. Retroalimentación**

- Cada acción da respuesta inmediata:

- Mensaje “Toma registrada”.
  - Vibración al confirmar.
  - Icono cambia de color.
- Errores muestran mensajes claros (“Debe ingresar la hora del recordatorio”).

## **6. Simplicidad**

- Flujo de tres pasos para agregar medicamento:
  1. Datos básicos
  2. Frecuencia
  3. Confirmación
- Las pantalla tienen solo lo esencial.
- Se evita mostrar información avanzada a usuarios básicos.



## 9.3 Diseño del prototipo (baja y alta fidelidad)

### Herramientas sugeridas

- Figma por la parte más profesional para realizar.

### Prototipos – Pantallas Principales



9:41

Meditrack





## Nueva cuenta

Llena el siguiente formulario con tus datos personales. Crea una contraseña y registra una cuenta de correo electrónico.

Nombre completo

Correo electrónico


Contraseña


Confirmar contraseña

Crear

9:41

Meditrack





# ¡Bienvenido!

Inicia sesión ahora

Correo electrónico

Contraseña

Ingresar

[Olvidé mi contraseña](#)



9:41 MedTrack

<

### Agregar Medicamento

Dosis

Frecuencia

Vía de administración

Hora

Duración


Agregar medicamento


9:41 MedTrack

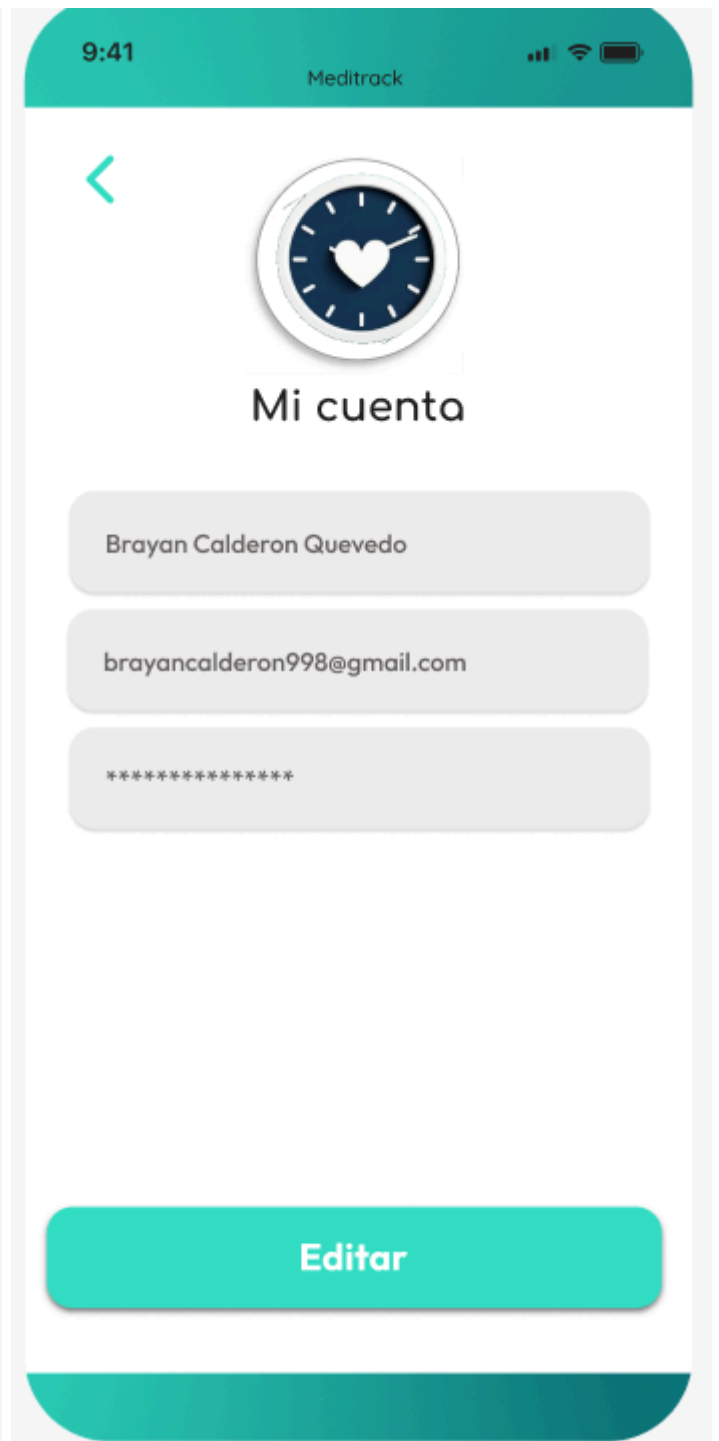
¡Bienvenido!

### Historial

Exportar PDF / Excel

Notificaciones 

Historial 



## **Justificación del diseño (colores, tipografías, organización)**

- **Colores**

- Verde (#32E0C4): éxito y confirmación.
- Gris (#EEEEEE): fondos limpios.

- **Tipografía**

- “Confortanaa”: altamente legible para adultos mayores.

- **Organización**

- Diseño modular para una lectura rápida.
- Jerarquía visual clara: lo más urgente arriba (próxima dosis).
- Botones grandes para accesibilidad.

## 9.4 Flujo de navegación del sistema

### Mapa de navegación (texto listo para diagrama)

Inicio → **Iniciar sesión** → Dashboard →

1. **Medicamentos**
  - ↳ Agregar medicamento
  - ↳ Editar medicamento
  - ↳ Eliminar medicamento
2. **Historial de tomas**
  - ↳ Filtrar
  - ↳ Exportar PDF/Excel
3. **Notificaciones**
  - ↳ Confirmar toma
  - ↳ Posponer
4. **Perfil**
  - ↳ Editar perfil
  - ↳ Configuración
  - ↳ Cerrar sesión


### Diagrama de interacción / módulo-navegación

- Usuario selecciona acción
- Sistema responde en menos de 2 segundos
- Validación → Procesamiento → Respuesta visual
- Para cada módulo:
  - **Dashboard → Medicamentos → Agregar → Guardar → Confirmación**
  - **Dashboard → Historial → Filtrar → Exportar**

- **Notificación → Confirmar/Posponer → Actualizar historial**

## 9.5 Relación entre prototipo y requerimientos

Aquí colocas una tabla como esta (lista para usar):

Pantalla	Requerimiento que satisface
Inicio de sesión	



## Registro de usuario

9:41 Meditrack

<

### Nueva cuenta

Llena el siguiente formulario con tus datos personales. Crea una contraseña y registra una cuenta de correo electrónico.

Nombre completo

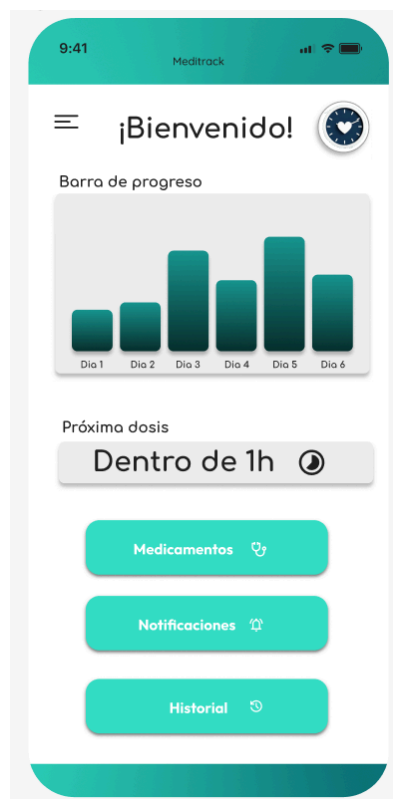
Correo electrónico

Contraseña

Confirmar contraseña

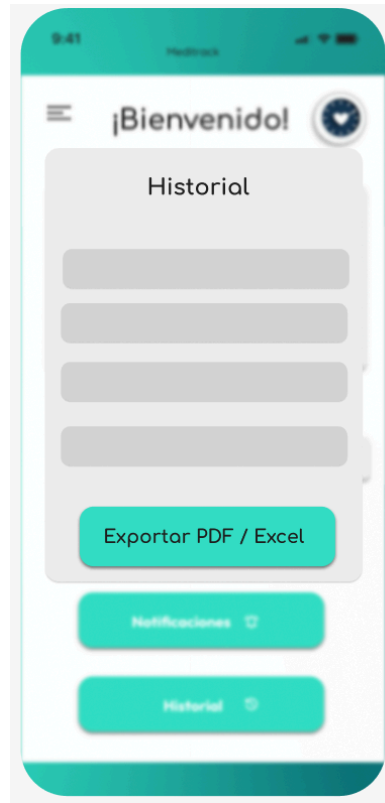
Crear

## Dashboard



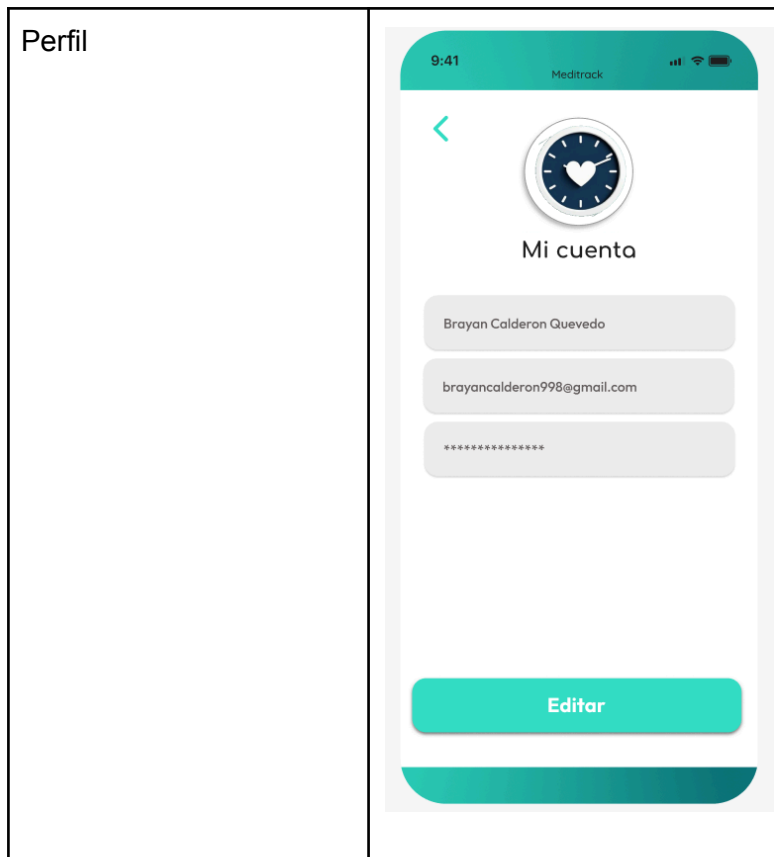
<p>Gestión de medicamentos</p>	
<p>Agregar medicamento</p>	

Historial



Notificaciones





Figma:

<https://www.figma.com/design/Ci0tCfDwlul3WjjfoTbqFj/Untitled?node-id=0-1&t=nHJzUJ6Suvp1jFPp-1>

# CAPÍTULO 10. Evaluación del Diseño y Matriz de Trazabilidad

## 10.1. Matriz de trazabilidad

Relacionar:

Req.	Descripción	Caso de uso	Clase o Módulo	Pantalla del prototipo	Cumple / Parcial / No cumple
RF1	El sistema debe permitir al usuario registrar medicamentos, especificando nombre genérico, dosis, frecuencia, vía de administración y duración del tratamiento.	(CS001)			
RF2	El sistema debe permitir enviar notificaciones y alarmas en tiempo real para recordar la toma de cada medicamento, incluso cuando el dispositivo se encuentre en modo silencioso.	(CS001)			
RF3	El sistema debe mostrar un historial de los medicamentos tomados y olvidados, y permitir exportar reportes en PDF o Excel.	(CS001)			

RF4	El sistema debe permitir la configuración de esquemas posológicos complejos, incluyendo excepciones, reprogramación temporal de dosis y función de posponer con límites definidos.	(CS001)			
RF5	El sistema debe permitir exportar y compartir de forma segura el historial de adherencia y efectos adversos en formatos estándar, con consentimiento explícito del paciente y registro en el historial de auditoría.	(CS001)			
RNF1	El sistema debe ser intuitivo y accesible para personas de todas las edades, asegurando una interfaz amigable y cumpliendo con las pautas de accesibilidad para personas con discapacidad visual o motriz.	(CS001)			
RNF2	El sistema debe responder a las acciones del usuario en menos de 2	(CS001)			

	segundos y consumir menos del 5% de la batería diaria del dispositivo.				
RNF3	El sistema debe cifrar toda la información médica y personal en tránsito y en reposo mediante protocolos seguros y gestión periódica de claves.	(CS001)			
RNF4	El sistema debe garantizar una disponibilidad mínima del 99 por ciento mensual en notificaciones y asegurar su entrega en un máximo de 60 segundos desde el horario programado.	(CS001)			

Debe incluir:

- RF (requerimientos funcionales)
- RNF (requerimientos no funcionales)
- Elementos del prototipo
- Elementos del diseño estructural y de arquitectura

#### 10.2. Evaluación del diseño del sistema

- Coherencia entre arquitectura, prototipo y base de datos.
- Identificación de mejoras y ajustes.
- Evaluación del cumplimiento de principios de usabilidad.

#### 10.3. Retroalimentación obtenida

Puede ser:

- Evaluación por pares
- Simulación de pruebas de usuario
- Revisión heurística de Nielsen

El estudiante debe describir:

- Errores detectados
- Cambios realizados
- Mejoras propuestas

#### 10.4. Reflexión sobre el aporte al ODS

Demostrar claramente cómo:

- El diseño de la interfaz,
- La navegación,
- Las funcionalidades

Contribuyen efectivamente con la ODS elegida.

## **CAPÍTULO 11. Consolidación Final y Evidencia de Trabajo Colaborativo (GitHub)**

### 11.1. Repositorio GitHub del proyecto

Debe contener carpetas para:

- /Documentacion
- /Modelos
- /BaseDatos
- /Prototipo
- /Trazabilidad
- /VersionesFinales

### 11.2. Evidencias de trabajo colaborativo

Colocar capturas de:

- Commits de cada integrante
- Branches • Pull Requests/Merges
- Issues creados y resueltos
- Insights → Contributors



### 11.3. Control de versiones de los documentos y modelos

- Mostrar cómo se fue actualizando el informe final.
- Etiquetar versiones: v1.0, v2.0, vFinal.

### 11.4. Reflexión del equipo

- Cómo GitHub mejoró la organización del proyecto
- Retos y aprendizajes del trabajo colaborativo
- Propuesta de mejoras futuras

## Conclusiones y Recomendaciones

- **Conclusiones del equipo:**

El sistema que desarrollamos representa un aporte significativo al ODS número 3 (Salud y bienestar) ya que ayuda a que los pacientes sigan mejor sus tratamientos y estén más saludables. La aplicación hace que sea más fácil para los médicos y los usuarios controlar las medicinas y recibir recordatorios a tiempo. Además, permite que cada persona tenga su propio perfil y que los médicos puedan ver cómo van sus pacientes.

- **Lecciones aprendidas:**

Durante el proyecto, aprendimos que trabajar en equipo y comunicarnos bien es muy importante para lograr buenos resultados. Revisar los requisitos desde el principio nos ayudó a evitar errores y a no perder tiempo. También vimos que usar herramientas digitales nos permitió organizarnos mejor y avanzar más rápido.

- **Recomendaciones:**

Para mejorar el sistema en el futuro, sugerimos conectarlo con relojes o pulseras inteligentes para que el control sea más automático. También sería bueno agregar un chat para que los pacientes puedan hablar fácilmente con sus médicos. Es importante hacer la aplicación más accesible para personas mayores o con alguna

discapacidad, y agregar alertas para farmacias sobre la renovación de recetas. Así, el sistema podrá ayudar a más personas y ser aún más útil.

## Referencias bibliográficas

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. *ISO/IEC 25010:2011 — Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models*. Ginebra: ISO, 2011.

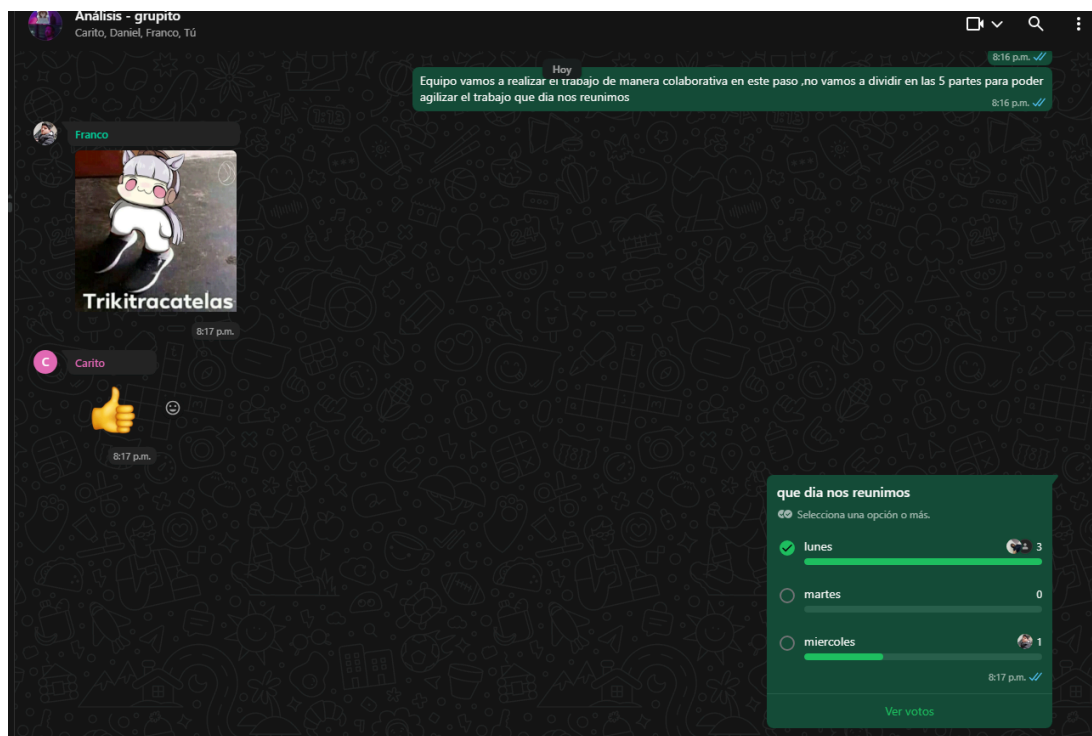
INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. *ISO/IEC 9126-1:2001 — Software engineering — Product quality — Part 1: Quality model*. Ginebra: ISO, 2001.

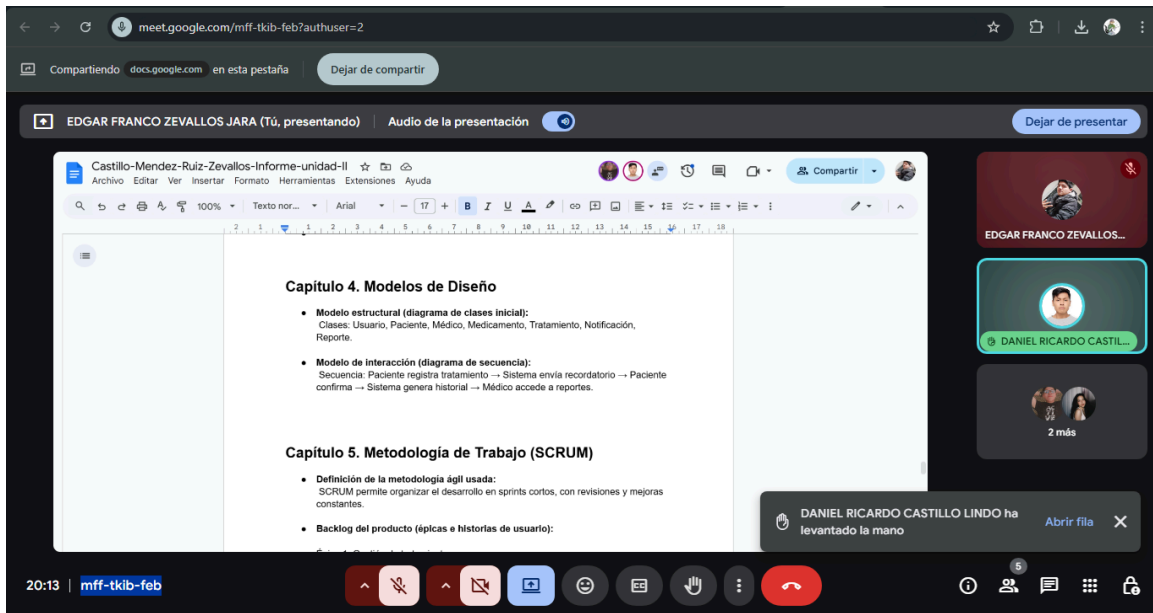
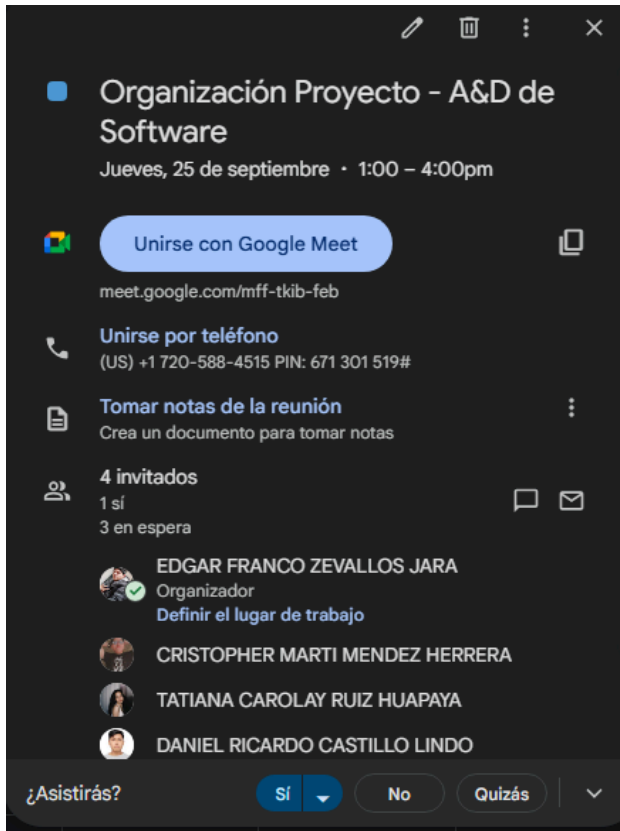
INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. *ISO/IEC 27001:2013 — Information technology — Security techniques — Information security management systems — Requirements*. Ginebra: ISO, 2013.

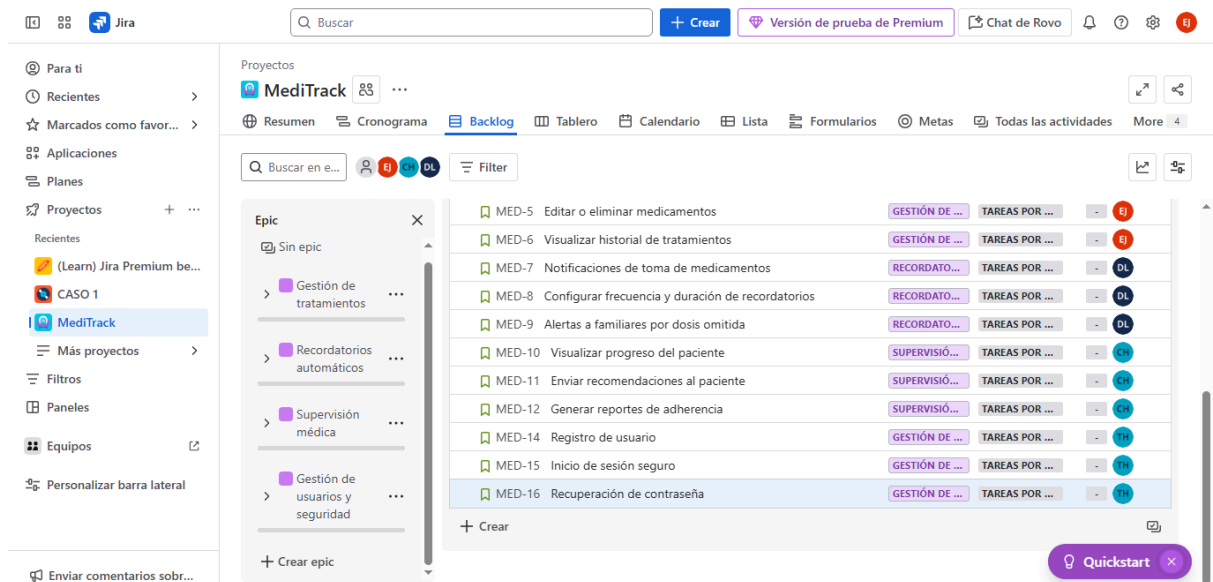
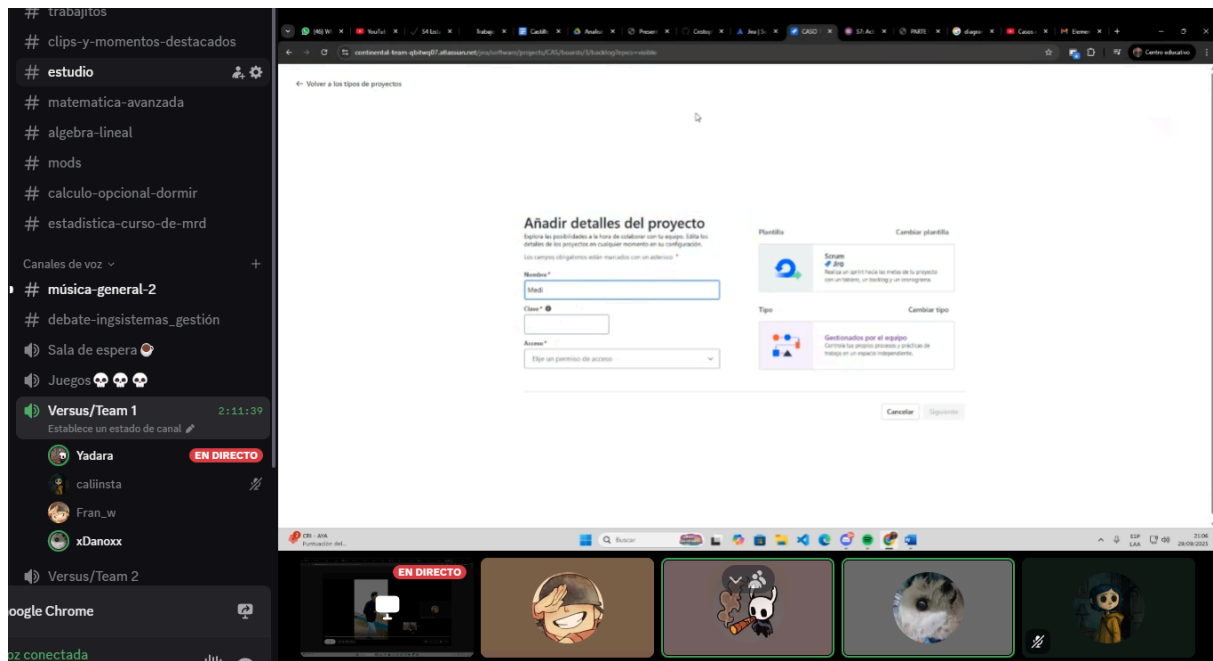
WORLD WIDE WEB CONSORTIUM. *Web Content Accessibility Guidelines (WCAG) 2.1*. 2018. Disponible en: <https://www.w3.org/TR/WCAG21/>. [Consulta: 30 de septiembre de 2025].

CONGRESO DE LA REPÚBLICA DEL PERÚ. *Ley N.º 29733 — Ley de Protección de Datos Personales*. Diario Oficial El Peruano, Lima, 2011.

## Anexos:











## Carga de trabajo del equipo

Supervisa la capacidad de tu equipo. [Reasignar actividades para obtener el equilibrio adecuado](#)

Persona asignada	Distribución del trabajo
 CRISTOPHER MARTI...	<div><div>25%</div></div>
 DANIEL RICARDO C...	<div><div>25%</div></div>
 EDGAR FRANCO ZE...	<div><div>25%</div></div>
 TATIANA CAROLAY ...	<div><div>25%</div></div>