



PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

Unidad Académica de Formación Técnica y Tecnológica – PUCE TEC

**APLICATIVO WEB PARA COORDINACIÓN DE CITAS MÉDICAS,
GESTIÓN DE PACIENTES E INFORMACIÓN DEL CENTRO DE
ESPECIALIDADES MÉDICAS SAÚDE**

**Proyecto de titulación previo a la obtención del título de: Tecnología
Superior De Desarrollo De Software**

Autor: Williams Andrés Sierra Narváez

Tutor: M.Sc.B.Eng. Jonathan David Quespaz Sánchez

Quito, Ecuador

2026

Dedicatoria

Esta tesis está dedicada:

A mí madre con mucho amor, Marlene Narváez Villota quién con su incondicional apoyo, paciencia y sobre todo cariño ha sido el pilar fundamental para que pueda cumplir esta meta, observar desde niño su perseverancia, esfuerzo y trabajo fue el impulso para que no me haya rendido en este camino y en los que me queda por afrontar.

A mis abuelos maternos, Cruz Elena Villota y Pánfilo Arnelio Narváez, en memoria de su vida y eterno amor, de quienes recibí todo el cariño, cuidado y apoyo desde que tengo memoria, lamento profundamente ya no estén presente físicamente para que pueda abrazarlos y darles la gracias, fuí muy afortunado en tenerlos.

A mis abuelos paternos, Carlos Bolívar Sierra en memoria de su vida y ejemplo de trabajo y Ana Luisa Argentina Guerrón, quienes con los frutos de su labor de toda la vida han sido un apoyo indispensable para llegar a cumplir este objetivo, mi cariño y gratitud por siempre.

A mis hermanas Kamyla Sierra Narváez, Diana Sierra Narváez y mi sobrino Steven Navarrete Sierra por ser mis compañeros de vida y parte de la motivación para cumplir esta meta.

A mis tías y tíos, mi agradecimiento por su apoyo constante, presencia y palabras de aliento.

Agradecimientos

A la Pontificia Universidad Católica del Ecuador, por haberme brindado una formación integral y los espacios necesarios para mi crecimiento académico y personal.

A los coordinadores de la carrera de software Ing. Diana Molina e Ing. Pablo Pérez, por su gestión, liderazgo y en especial por su amistad.

A mi tutor, Ing. Jonathan Quespaz, por su tiempo, paciencia y valiosa orientación en el desarrollo de este trabajo.

A todos mis docentes, por compartir su conocimiento y compromiso en la enseñanza.

Al personal de la Dirección de Informática de la PUCE, Por su amistad y haberme recibido durante mi periodo de prácticas preprofesionales. Agradezco la oportunidad de aprender de su experiencia en el campo laboral.

Índice

Dedicatoria.....	2
Agradecimientos	3
Declaración y autorización	10
Introducción.....	11
Definición y Planificación del Proyecto Tecnológico	12
Planteamiento del Problema y Argumentación.....	12
1.1 Objetivos del Proyecto.....	13
1.2 Objetivo General	13
1.3 Objetivos Específicos.....	14
Metodología de Desarrollo.....	15
1.1 Enfoque de Investigación.....	15
1.2 Estructura metodológica ágil SCRUM	15
1.3 Herramientas de Apoyo	16
1.4 Marco Conceptual.....	17
1.4.1 Desarrollo Front-End:	17
1.4.2 Desarrollo Back-End:.....	17
1.4.3 Gestión de Datos:	17
1.4.4 Seguridad del Sistema:.....	18
CAPITULO I	19
Marco Conceptual.....	19
1. Procesos Actuales (Situación AS-IS)	20

2. Marco de Trabajo.....	21
3. Adaptación de Roles en el Desarrollo.....	21
3.1 Ceremonias de Gestión	22
Sprint I	23
Sprint II	24
Sprint III.....	25
Sprint IV	26
CAPITULO II.....	27
Análisis y Diseño	27
1.1 Administrador	27
1.2 Médico: Su entorno de visualización es más limpio sin actividades administrativas.....	28
Justificación Técnica.....	28
1.1 Centralización y Consistencia de la Información Clínica.....	28
1.2 Automatización de la Logística de Atención	29
1.3 Seguridad y Responsabilidades	29
Diseño de Arquitectura de Software	30
1.1 Selección del Patrón Arquitectónico.....	30
1.2 Capa de Presentación (Frontend - Cliente)	30
Diagrama de Navegación	31
1.3 Backend Lógica Servidor.....	32
Acceso a Datos (ORM - Intermediario).....	33

Base de Datos.....	33
Diagrama de Arquitectura de Software.....	34
Gráfico de Inicio de sesión de Roles, Seguridad y Base de Datos.....	35
Análisis de Viabilidad Técnica	36
1.1 Requisitos para Estaciones de Trabajo	36
1.2 Análisis de Despliegue.....	38
CAPITULO III	39
Implementación.....	39
1. Alcance	39
1.1 Módulo de Agendamiento Inteligente.....	39
1.2 Sistema de Notificaciones y Recordatorios	39
1.3 Administración de Pacientes y Especialidades	40
1.4 Segregación de Paneles (Administrador, Médico).....	40
2. Arquitectura Cliente – Servidor	41
2.1 Arquitectura y Flujos de Comunicación	41
2.2 Organización Modular del Sistema.....	42
3. ESTÁNDAREZ DE INTERFAZ Y USABILIDAD	44
3.1 Estándares de Uso y Experiencia de Usuario (UX)	44
3.2 Accesibilidad Web (WCAG AA) y Principios POUR.....	44
3.3 Aplicación de las Heurísticas de Nielsen.....	46
3.4 Diseño Centrado en la Experiencia de usuario	47
4. GESTIÓN DE PACIENTES	47

.....	47
5. Módulos funcionales.....	48
5.1 Módulos para Gestión de Citas	48
5.2 Catálogo de Especialidades Médicas	49
5.3 Panel Administrativo Centralizado	50
5.4 Panel Médico.....	51
5.5 Optimización del Tiempo Operativo.....	51
6. Recordatorios	52
6.1 Automatización de Notificaciones y Recordatorios.....	52
6.2 La lógica de notificación:.....	52
7. Seguridad	54
7.1 Implementación de Seguridad y Control de Acceso	54
7.2 Autenticación y Gestión de Sesiones (JWT).....	54
7.3 Despliegue Seguro (HTTPS)	55
7.4 Política de Primer Acceso y Cambio de Clave	55
CONCLUSIONES.....	57
RECOMENDACIONES	58
REFERENCIAS BIBLIOGRÁFICAS	59
ANEXOS.....	60
LISTA DE TABLAS	60
LISTA DE FIGURAS.....	65
.....	71

Índice de Tablas

Tabla 1	16
Tabla 2	23
Tabla 3	24
Tabla 4	25
Tabla 5	26
Tabla 6	37
Tabla 7	38

Índice de Ilustraciones

Ilustración 1	13
Ilustración 2	20
Ilustración 3	21
Ilustración 4	31
Ilustración 5	32
Ilustración 6	34
Ilustración 7	35
Ilustración 8	43
Ilustración 9	44
Ilustración 10	45
Ilustración 11	46
Ilustración 12	47
Ilustración 13	48
Ilustración 14	49
Ilustración 15	50
Ilustración 16	51
Ilustración 17	56

Declaración y autorización

Yo, Williams Andrés Sierra Narváez con C.I. 0401876032 autor del trabajo de Integración curricular intitulado: APLICATIVO WEB PARA COORDINACIÓN DE CITAS MÉDICAS, GESTIÓN DE PACIENTES E INFORMACIÓN DEL CENTRO DE ESPECIALIDADES MÉDICAS SAÚDE

Previa a la obtención del título de Tecnología Superior De Desarrollo De Software en la Unidad Académica de Formación Técnica y Tecnológica PUCE TEC:

1.- Declaro tener pleno conocimiento de la obligación que tiene la Pontificia Universidad Católica del Ecuador, de conformidad con el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de graduación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la Pontificia Universidad Católica del Ecuador a difundir a través de sitio web de la Biblioteca de la PUCE el referido trabajo de titulación, respetando las políticas de propiedad intelectual de Universidad.

Quito, 22 de noviembre de 2025



Williams Andrés Sierra Narváez

C.I. 0401876032

Introducción

El sector salud se enfrenta al reto de organizar de manera eficiente sus procesos administrativos y de atención, especialmente en centros médicos con diversas especialidades, donde la falta de herramientas digitales puede generar demoras en la asignación de citas, errores en los registros de pacientes y dificultades en la difusión de servicios. Este aspecto afecta la calidad del servicio y también la seguridad de los datos médicos, al no existir mecanismos adecuados para su resguardo.

Para atender estas necesidades, la propuesta es desarrollar una aplicación web para el Centro Médico Saúde que funciona sobre tres pilares: gestión de citas médicas, registro digital de pacientes y consulta de especialidades con sus respectivos profesionales.

La intención es entregar al equipo administrativo y médico un sistema rápido que reduzca los tiempos al agendar las consultas, minimice los errores en el registro y que haya la garantía de seguridad y confidencialidad de los datos.

Por otro lado, buscamos dar a los administradores del centro médico un mejor control de trabajo interno, mediante un sistema de roles y permisos. Básicamente se busca construir este sistema para elevar la calidad del servicio que ofrece centro médico y optimizar el tiempo de sus trabajadores.

Definición y Planificación del Proyecto Tecnológico

Planteamiento del Problema y Argumentación.

Gestionar la información en centros de salud requiere precisión, inmediatez y seguridad para ofrecer garantía de una atención de calidad. En el caso del Centro Médico Saúde, se identificó que la operatividad actual depende de procesos manuales y canales informales como redes sociales para el agendamiento de citas. Esta situación genera inconsistencias en la disponibilidad de horarios, duplicidad de registros y una percepción poco profesional ante el paciente.

La transformación digital en el sector salud no es solo una tendencia, sino una necesidad operativa. Según (Organización Panamericana de la Salud, 2021) la implementación de sistemas automatizados reduce significativamente los errores humanos y optimiza los tiempos de atención administrativa. Por tanto, el presente proyecto técnico-tecnológico se justifica en la necesidad de migrar de un modelo de gestión manual (AS-IS) a un modelo automatizado e integrado (TO-BE), centralizando la información de pacientes, médicos y citas en una plataforma web única.

Figura 1. Proceso de actual de operaciones administrativas

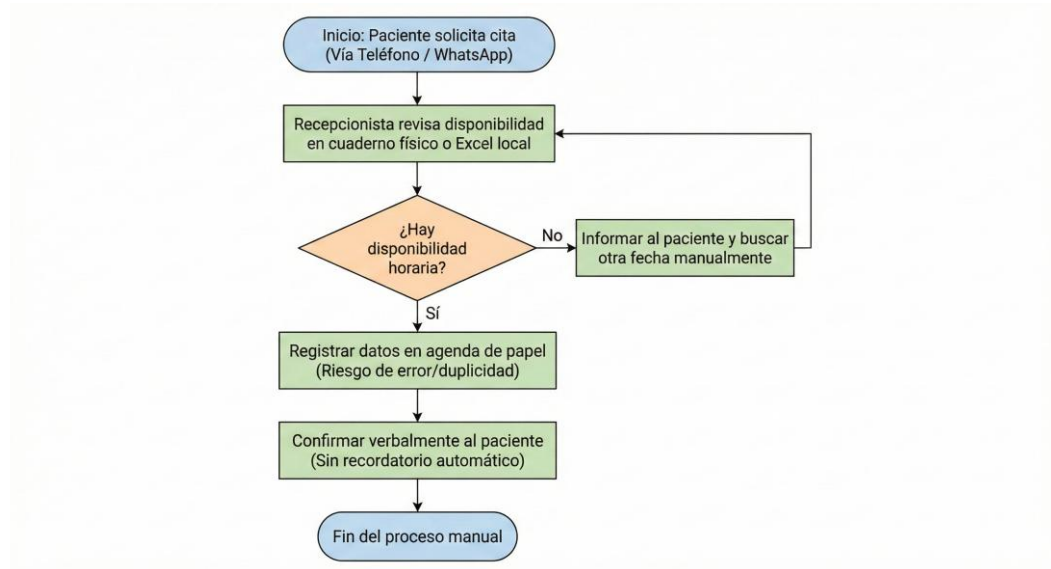


Ilustración 1

- Nota. La figura del diagrama de flujo muestra los múltiples pasos manuales requeridos para confirmar una sola cita.

1.1 Objetivos del Proyecto

Para dar solución a la problemática planteada y garantizar la coherencia del desarrollo, se establecieron los siguientes objetivos que guían la construcción de los módulos de software.

1.2 Objetivo General

Formular e implementar un aplicativo web para el Centro Médico Saúde que unifique el agendamiento de citas, el registro de pacientes y la consulta de especialidades, con el fin de optimizar la gestión, experiencia del usuario e imagen profesional de la empresa.

1.3 Objetivos Específicos

La operatividad detallada del proyecto se plantea en los siguientes objetivos técnicos:

- a. Precisar el alcance del proyecto, incluyendo el agendamiento, recordatorios de citas, administración básica de pacientes, publicación de especialidades y panel administrativo y Médico.
- b. Definir la arquitectura cliente–servidor con API REST, dejando claros los módulos y flujos básicos de comunicación.
- c. Asegurar que la interfaz cumpla estándares reconocidos de usabilidad y accesibilidad (WCAG AA) con principios POUR y de Nielsen, para lograr una óptima Experiencia de Usuario (UX).
- d. Registrar y gestionar pacientes (agregar, edición, eliminación) aplicando validaciones de negocio como verificación de cédula, nombre y correo electrónico.
- e. Implementar los módulos funcionales de citas, catálogo de especialidades y panel administrativo, orientados a optimizar el tiempo de agendamiento.
- f. Configurar recordatorios automáticos por correo electrónico para confirmación y aviso 24 horas antes de la cita médica.
- g. Aplicar seguridad esencial mediante inicio de sesión con JWT, segregación de roles (administrativo, médico) y uso de HTTPS en el despliegue.

Metodología de Desarrollo

1.1 Enfoque de Investigación

Para la construcción del sistema se adoptó un enfoque mixto. En el componente cualitativo, se realizaron diálogos y observación directa con el personal del centro médico para levantar las necesidades reales y acordar los criterios de aceptación. En el componente cuantitativo, se estableció una línea base midiendo el tiempo promedio de agendamiento y el porcentaje de errores en registros manuales, con el fin de contrastar estos indicadores tras la implementación del software.

1.2 Estructura metodológica ágil SCRUM

La gestión del proyecto se fundamentó en Scrum. Este es un marco de trabajo ligero que ayuda a las personas y equipos a generar valor a través de soluciones adaptables para problemas complejos. (Schwaber & Sutherland, 2020)

La operativa se estructuró en iteraciones denominadas Sprints. Los Sprints son eventos de duración fija donde las ideas se convierten en valor y funcionan como el ritmo cardíaco (heartbeat) de Scrum (Schwaber & Sutherland, 2020).

El ciclo de desarrollo siguió estos pasos:

- a. Planificación Sprint Planning: Se definió el trabajo a realizar seleccionando ítems del Product Backlog, el cual es una lista emergente y ordenada de lo que se necesita para mejorar el producto (Schwaber & Sutherland, 2020).
- b. Ejecución: Durante el Sprint, se transforman los requerimientos en incrementos de software funcional.

- c. **Revisión:** Cuando finaliza cada ciclo, se revisa el funcionamiento real del software y se ajusta el plan para la siguiente etapa, así se asegura que los resultados son siempre visibles y claros para evitar errores o desviaciones.

1.3 Herramientas de Apoyo

Para garantizar la calidad técnica, la colaboración y la trazabilidad del desarrollo, se utilizaron las siguientes herramientas basadas en documentación técnica:

- **Control de Versiones Git:** Se empleó Git, definido como un sistema de control de revisiones distribuido, rápido y escalable (Git, n.d.) para gestionar el historial de cambios del código fuente y permitir flujos de trabajo ramificados seguros.
- **Gestión de Repositorios GitHub:** El código se alojó en GitHub, una plataforma basada en la nube que permite almacenar, compartir y trabajar en conjunto con otros usuarios para escribir código (GitHub, n.d.) facilitando además el seguimiento de tareas mediante su sistema de Issues.

Tabla 1

Componente	Elemento	Aplicación en el Proyecto
Enfoque de Investigación	Cualitativo	Diálogos y observación directa para levantar necesidades y definir criterios de aceptación.
	Cuantitativo	Establecimiento de línea base (tiempos y errores) para contrastar resultados post-implementación.
Marco de Trabajo (Scrum)	Sprints	Iteraciones de duración fija que funcionan como el "ritmo cardíaco" para entregar valor continuo.
	Planificación	Selección de tareas desde el Product Backlog (lista ordenada de necesidades).
	Ejecución	Transformación de requerimientos en incrementos de software funcional.
	Revisión	Inspección del funcionamiento real para ajustar el plan y asegurar transparencia.
Herramientas de Apoyo	Git	Sistema de control de versiones distribuido para gestionar el historial de cambios.
	GitHub	Plataforma en la nube para alojamiento de código y gestión de tareas mediante <i>Issues</i> .

1.4 Marco Conceptual

Para asegurar que la plataforma sea funcional, segura y fácil de mantener, el desarrollo se apoya en los siguientes pilares tecnológicos y operativos:

1.4.1 Desarrollo Front-End:

Se utiliza React para construir una interfaz rápida y modular, permitiendo que las pantallas de agendamiento y gestión de pacientes respondan de inmediato sin recargar la página. El diseño visual se apoyará en Tailwind CSS, lo que garantiza que el sistema se vea bien y sea fácil de usar tanto en monitores de escritorio de la recepción como en dispositivos móviles.

1.4.2 Desarrollo Back-End:

La lógica del servidor se gestiona con **Node.js** y el framework **Express**. Esta combinación se encargará de procesar las reglas, como verificar que un médico no tenga dos citas a la misma hora y controlar el flujo de datos entre el usuario final y el sistema de base de datos mediante una API REST.

1.4.3 Gestión de Datos:

Se implementa PostgreSQL como gestor principal de la base de datos debido a su robustez para manejar información relacional crítica, como los vínculos entre pacientes, médicos y sus citas. Para interactuar con estos datos de forma segura, se usa Prisma ORM, una herramienta que valida los tipos de datos antes de guardar cualquier registro, previniendo errores humanos en la estructura de la información.

1.4.4 Seguridad del Sistema:

En lugar de sesiones tradicionales, se implementará autenticación basada en JWT (JSON Web Tokens). Esto permite crear un sistema de permisos donde por ejemplo el "Administrador" tiene acceso total y el "Médico" solo ve su agenda, asegurando que nadie acceda a información que no le corresponde.

CAPITULO I

Marco Conceptual

Hoy en día, la eficiencia en el sector de la salud no es solo una ventaja competitiva, es un aspecto básico para dar una atención de la calidad. Por eso la transformación digital en centros de salud ya no es opcional, es necesaria.

Automatizar tareas administrativas, como agendar las citas médicas, o asignar médicos es una estrategia para acortar tiempos de espera y reducir los errores humanos que ocurren en una transcripción manual de datos.

Al investigar el ambiente local, grandes hospitales públicos o clínicas privadas ya manejan sistemas informáticos robustos, los centros médicos pequeños y medianos, como el Centro Médico Saúde, suelen operar con procesos manuales divididos, agendas en hojas físicas o de cálculo, mensajes vía chat o correos individuales enviados a mano uno por uno, lo que dificulta y desordena los procesos administrativos y compromete la seguridad de los datos de los pacientes ante un posible error humano.

La problemática central se enfoca en que los sistemas manuales que no permiten una gestión proactiva. Por ejemplo, la confirmación de citas depende de la memoria del recepcionista, notas físicas, llamadas telefónicas que muchas veces no son contestadas, derivando en una alta tasa de ausentismo. En este sentido, el desarrollo de un sistema web a medida no solo busca digitalizar el papel, sino reestructurar la manera de trabajar para que el personal médico y administrativo pueda centrarse en la atención al paciente en lugar de la burocracia y cabe mencionar la profesionalidad que tener una web le da a una empresa orientada a la prestación de servicios, como en este caso.

1. Procesos Actuales (Situación AS-IS)

A continuación, se describe los flujos operativos vigentes en el Centro Médico Saúde. Estos procesos se caracterizan por una alta dependencia de las hojas de cálculo y falta de sincronización entre recepción y los consultorios médicos.

Proceso manual de agendamiento de citas en Saúde

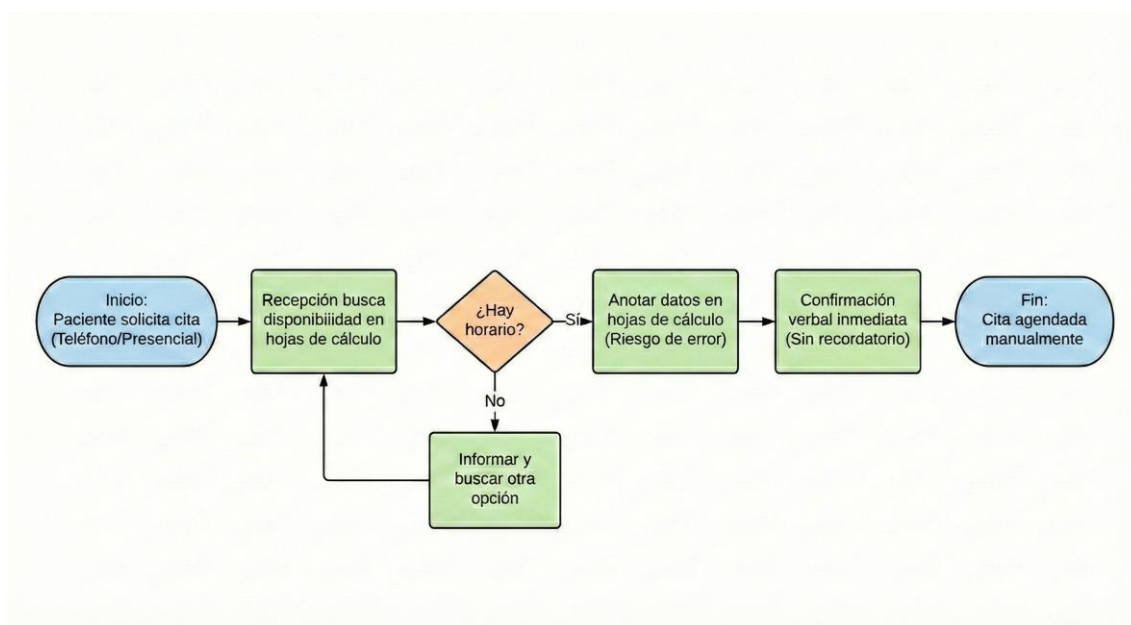


Ilustración 2

Nota. El diagrama ilustra el flujo actual donde la disponibilidad se verifica en hojas de cálculo, generando riesgo de duplicidad de horarios

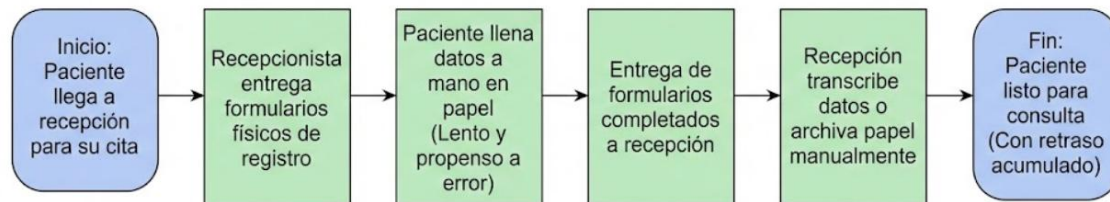


Ilustración 3

Nota. Representación del llenado de formularios en papel previo a la consulta, lo que retrasa el tiempo de atención efectiva.

2. Marco de Trabajo

Para el desarrollo del sistema web "Saúde", se seleccionó Scrum como marco de trabajo ágil. La elección está justificada en la necesidad de gestionar un desarrollo iterativo donde los requisitos definidos inicialmente, es necesario una refinación constante tras la validación con el personal médico. Scrum nos permite dar valor a con resultados que se adaptan para resolver problemas más delicados, lo cual se alinea con el objetivo de transformar procesos manuales en digitales.

3. Adaptación de Roles en el Desarrollo

Dado que el proyecto fue ejecutado por un único desarrollador, los roles tradicionales de Scrum se adaptaron para concentrar las responsabilidades del marco de trabajo:

- **Product Owner (Dueño del Producto):** Este rol implicó la interlocución directa con el Centro Médico Saúde para priorizar las funcionalidades críticas (agendamiento y gestión de pacientes) sobre las deseables.
- **Scrum Máster:** Enfocado en la autogestión, asegurando el cumplimiento de los tiempos establecidos para cada Sprint y eliminando bloqueos técnicos mediante la investigación de documentación (Prisma, React).
- **Desarrollador:** Rol ejecutado por Andrés Sierra. Análisis, diseño, codificación Full Stack y pruebas de calidad.

3.1 Ceremonias de Gestión

Para mantener el ritmo de trabajo ("heartbeat"), se respetaron los hitos temporales de Scrum adaptados al desarrollo individual:

- **Sprint Planning:** Para comenzar un ciclo, se seleccionaron las historias de usuario a desarrollar (ej. "Como médico quiero ver mi agenda").
- **Ejecución y Daily (Auto-seguimiento):** Revisión diaria del progreso de código y actualización de tareas localmente.
- **Sprint Review & Retrospective:** Al concluir un módulo, se verifica la funcionalidad (ej. que la cita se agendara correctamente en el administrador y medico) y se ajustaron las técnicas de programación para el siguiente ciclo.

Planificación de Sprints

El cronograma de desarrollo se estructuró 4 Sprints con una duración promedio de tres semanas cada uno, cada sprint está enfocado en tareas específicas de desarrollo en frontend, backend, requerimientos e interacción con Saúde.

Sprint I

En esta primera etapa se realizaron las visitas presenciales para entender la operación del centro médico y se preparó toda la base técnica necesaria para comenzar la programación.

Tabla 2

Tarea	Responsable	Duración
Reunión de inicio con la dirección médica para definir el alcance del proyecto.	Andrés Sierra	2 días
Observación en sitio del proceso manual de agendamiento y atención al paciente.	Andrés Sierra	3 días
Inspección técnica de los equipos de cómputo e internet del centro médico.	Andrés Sierra	1 día
Redacción y validación de la lista de requerimientos del sistema.	Andrés Sierra	2 días
Definición de la arquitectura de la base de datos (tablas de administrador, médicos, pacientes, citas).	Andrés Sierra	3 días
Preparación del entorno de desarrollo (local).	Andrés Sierra	2 días
Creación del repositorio de código y estructura de carpetas del proyecto.	Andrés Sierra	1 día
Ajuste inicial de la base de datos y enlace con el servidor.	Andrés Sierra	1 día

Sprint II

Se construyó toda la parte visual del sistema (Frontend), permitiendo que los usuarios interactúen con la base de datos a través de una interfaz gráfica amigable, intuitiva

Tabla 3

Sprint II: Desarrollo del Frontend y Experiencia de Usuario	Duración: 3 Semanas	
Tarea	Responsable	Duración
Configuración inicial del proyecto visual y definición de la paleta de colores del centro médico.	Andrés Sierra	2 días
Diseño y programación de componentes reutilizables (Botones, campos de texto, tarjetas de información).	Andrés Sierra	2 días
Desarrollo de la vista de iniciar de sesión (Login) e integración con la seguridad del servidor.	Andrés Sierra	2 días
Construcción del menú de navegación lateral.	Andrés Sierra	2 días
Desarrollo de la interfaz para el listado de pacientes.	Andrés Sierra	2 días
Implementación de los formularios de registro y edición de historias clínicas.	Andrés Sierra	2 días
Diseño de la vista principal del calendario para la gestión visual de citas.	Andrés Sierra	3 días
Conexión de las pantallas del módulo administrativo con la base de datos.	Andrés Sierra	2 días
Desarrollo visual Calendario de citas.	Andrés Sierra	1 día
Conexión de rutas entre panel Administrativo y médico.	Andrés Sierra	1 día

Sprint III

Se abordan las funcionalidades más complejas del sistema, combinando lógica en el servidor con componentes visuales dinámicos para la gestión del tiempo de los médicos.

Tabla 4

Sprint III: Funcionalidades Críticas y Gestión de Agenda	Duración: 3 Semanas	
Tarea	Responsable	Duración
Desarrollo del endpoint y lógica para detectar el primer inicio de sesión del usuario.	Andrés Sierra	1 día
Implementación del flujo de cambio de contraseña obligatorio por seguridad en el primer acceso.	Andrés Sierra	2 días
Programación del algoritmo de validación en el servidor para impedir choques de horarios entre citas.	Andrés Sierra	3 días
Integración y personalización de la librería de calendario interactivo en el Frontend.	Andrés Sierra	2 días
Desarrollo de la vista "Mi Agenda" filtrada exclusivamente para el usuario con rol de Médico.	Andrés Sierra	3 días
Programación de la interactividad del calendario: selección de bloques horarios y modales de detalle.	Andrés Sierra	3 días
Conexión de la interfaz de agendamiento con la base de datos (Carga y guardado de citas).	Andrés Sierra	2 días
Implementación de la funcionalidad para cancelar o reprogramar citas desde el calendario.	Andrés Sierra	2 días
Ajustes visuales y de usabilidad en la visualización de la agenda por día, semana y mes.	Andrés Sierra	1 día

Sprint IV

Fase final se implementa la automatización de comunicaciones para reducir el ausentismo de pacientes y se dejó el sistema completamente operativo para su uso real.

Tabla 5

Sprint IV: Notificaciones, Módulo Público y Entrega Final	Duración: 3 Semanas	
Tarea	Responsable	Duración
Configuración del servicio de envío de correos (usando cuenta Gmail y SMTP) en el servidor.	Andrés Sierra	2 días
Programación de la notificación automática al paciente inmediatamente después de agendar una cita.	Andrés Sierra	2 días
Implementación de la tarea automática diaria (6:00 AM) para enviar recordatorios de citas de las próximas 24 horas.	Andrés Sierra	3 días
Mejora de la sección "Especialidades" en la página principal (Landing Page) para información pública.	Andrés Sierra	3 días
Pruebas integrales del sistema: validar que los correos electrónicos lleguen a la bandeja de entrada en lugar de la bandeja spam.	Andrés Sierra	2 días
Corrección de detalles estéticos finales y errores menores reportados en las pruebas.	Andrés Sierra	2 días
Preparación del entorno final de despliegue.	Andrés Sierra	1 día

CAPITULO II

Análisis y Diseño

Actualmente, el Centro de especialidades médicas Saúde carece de una plataforma tecnológica integrada que centralice la gestión de sus procesos administrativos. La operativa diaria que abarca desde la programación de citas hasta el registro de pacientes nuevos, depende casi exclusivamente de herramientas ofimáticas aisladas como hojas de cálculo o bitácoras físicas. Esta desconexión provoca que no exista una validación en tiempo real de la disponibilidad de los médicos o el registro de pacientes que se basa en métodos antiguos de seguimiento,

Tras realizar un diagnóstico operativo, se detectaron cuellos de botella significativos que afectan la calidad del servicio: duplicidad de horarios por errores, una comunicación Administrador – Médico vía canales de chat personal como WhatsApp, recurrentes inasistencias de pacientes por no tener un sistema de recordatorios de citas y errores humanos característicos a la hora de asignar pacientes a una especialidad, choque de horarios etc.

Ante esta realidad, se propone el desarrollo de un sistema web de gestión médica que modernice y optimice estos flujos de trabajo. La solución busca reemplazar los archivos dispersos por una base de datos relacional robusta (PostgreSQL), garantizando la integridad de la información y permitiendo el acceso concurrente y seguro mediante autenticación JWT.

Para organizar el ingreso a la plataforma, se usa un sistema de permisos que garantiza que cada rol de usuario acceda solo a lo necesario para su trabajo para mantener simplicidad y seguridad, los roles se dividen en:

1.1 Administrador: Este es el rol maestro, sus tareas consisten en crear y configurar las cuentas de nuevo personal médico, configura horarios desde su panel,

tienen acceso a los pacientes de cada usuario médico y se encarga de registrar pacientes nuevos, es un sistema escalable que posteriormente podría generar reportes a partir de los datos recopilados.

1.2 Médico: Su entorno de visualización es más limpio sin actividades administrativas. Podrá visualizar su agenda personal en tiempo real (diaria, semanal o mensual), acceder a la información de contacto de sus pacientes, gestionando citas a partir de una segunda asignación para seguimiento o tratamientos prolongados, eliminando la necesidad de consultar a recepción sobre su programación del control de pacientes recurrentes.

Justificación Técnica

El levantamiento de información realizado en el Centro Médico Saúde permitió identificar brechas críticas entre la operatividad manual actual y los estándares de gestión clínica modernos. El análisis no solo se centra en la digitalización, sino en la reingeniería de los flujos de información para asegurar la integridad de los datos y la optimización de los recursos humanos.

1.1 Centralización y Consistencia de la Información Clínica

El uso actual de herramientas ofimáticas desconectadas (hojas de cálculo y agendas físicas) impide la concurrencia de usuarios y carece de integridad referencial. Esto significa que no existen controles que eviten la duplicidad de registros o la inconsistencia en los datos de los pacientes. El nuevo sistema debe actuar como una fuente única. Se requiere una base de datos relacional que imponga reglas estrictas de negocio: validación de disponibilidad en tiempo real, vinculación inquebrantable entre médico y cita, y seguridad en el acceso a la información sensible, algo que los archivos compartidos u hojas físicas no pueden ofrecer.

1.2 Automatización de la Logística de Atención

Se identificó que una parte significativa del tiempo operativo del personal de recepción se invierte en tareas repetitivas de bajo valor, específicamente la confirmación vía chat de asistencias. Este método manual es insostenible a medida que crece el volumen de pacientes y tiene un impacto directo en la tasa de ausentismo (pacientes que olvidan su cita). El requisito funcional es eliminar esta dependencia manual. El sistema debe incorporar un motor de notificaciones autónomo, que gestione el ciclo de vida de la cita sin intervención humana, enviando comprobantes inmediatos y recordatorios preventivos (24 horas antes). Esto mejora la experiencia del usuario paciente, sino que recupera horas productivas del personal para la atención presencial.

1.3 Seguridad y Responsabilidades

A diferencia del manejo actual donde el acceso a la información es indiscriminado para quien tenga el archivo físico o digital, el sistema debe implementar una arquitectura de seguridad basada en roles. Es imperativo que la gestión administrativa (configuración global, gestión de usuarios) esté estrictamente separada de la operativa médica (agenda personal, datos de pacientes), garantizando la privacidad y el seguimiento de normas éticas en el manejo de datos de salud.

Diseño de Arquitectura de Software

La arquitectura de software constituye, un centro médico donde la precisión de los datos y la disponibilidad inmediata es de suma importancia, no se podía optar por una estructura rígida monótona. Por eso decidimos construir sobre una arquitectura de modelo Cliente-Servidor. Esta elección tiene una ventaja práctica enorme, nos permite hacer mejoras o arreglos en el servidor sin tener que ir computadora por computadora actualizando dispositivos de cada médico, recepcionista o administrador.

1.1 Selección del Patrón Arquitectónico

El pilar del sistema funciona como una API RESTful. Esto significa que separamos totalmente lo que ve el usuario (Frontend) de donde viven los datos (Backend); ambos son independientes y solo "hablan" entre sí mediante peticiones HTTP seguras.

1.2 Capa de Presentación (Frontend - Cliente)

Esta es la cara visible para los administradores, médicos y pacientes. A diferencia de los sistemas web antiguos donde el servidor tenía que "dibujar" las pantallas una a una cada vez que se daba un clic, en Saúde usamos React para crear lo que se conoce como una SinglePage Application.

¿Qué ganamos con esto? Que toda la carga visual la procese el navegador del usuario, así se da una experiencia muy fluida, parecida a usar un programa de escritorio instalado. limitándose a capturar lo que hace el usuario y mostrar datos, pero dejamos claro que esta capa no toma decisiones ni procesa las reglas pesadas."

Diagrama de Navegación

Figura 4

Nota: Diagrama Navegación en el front-end, que ilustra la segregación de interfaces y permisos de acceso para los roles de administrador y médico tras la autenticación.

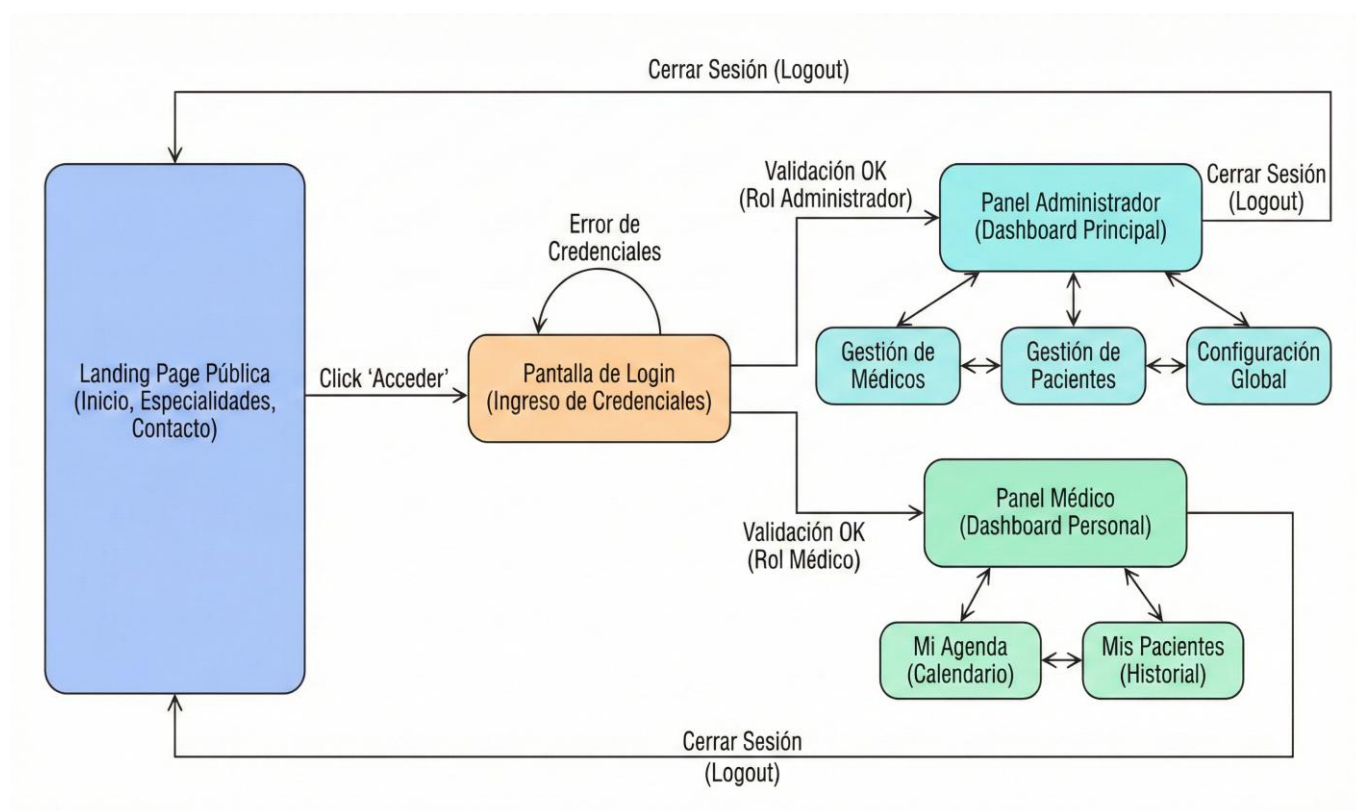


Ilustración 4

1.3 Backend Lógica Servidor

Constituye el núcleo inteligente del sistema. Desarrollada sobre Node.js y Express, esta capa actúa como el guardián de las reglas operativas del centro médico. Aquí es donde reside la programación que impide el agendamiento de citas duplicadas, gestiona la encriptación de contraseñas y orquesta el envío de correos electrónicos. Esta capa es "ciega" a la interfaz; no sabe si la petición viene de un celular o una computadora, simplemente recibe datos, los valida estrictamente y devuelve una respuesta estructurada (JSON).

Diagrama de lógica interna del servidor.

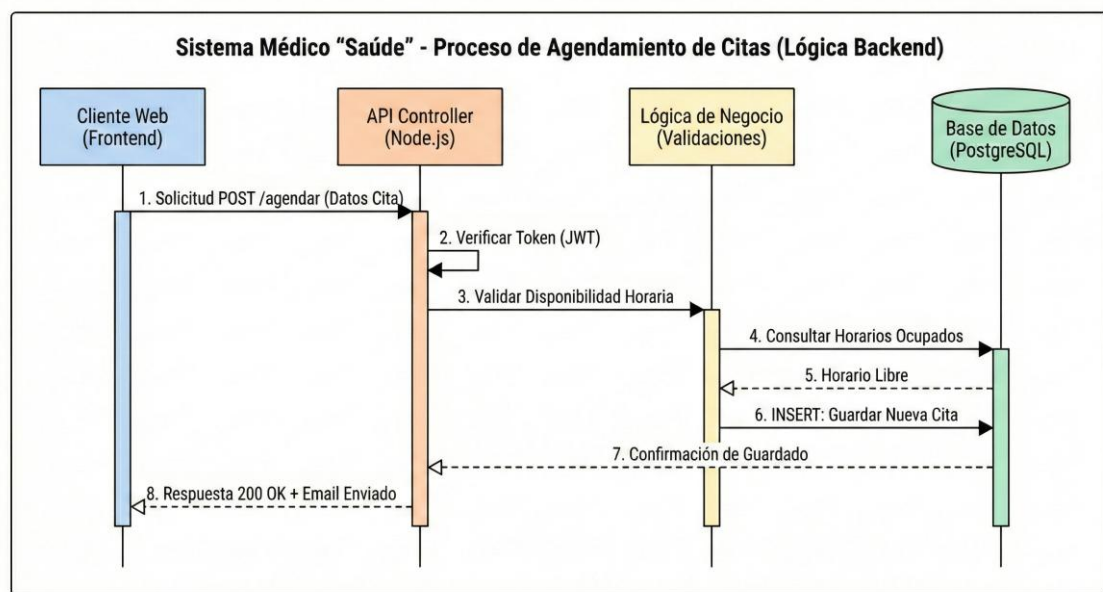


Ilustración 5

Nota: Diagrama que detalla la lógica interna en el servidor del sistema.

Acceso a Datos (ORM - Intermediario)

Para asegurar la integridad de la información médica y prevenir errores comunes en consultas SQL manuales, se implementa Prisma ORM. Este componente funciona como un traductor seguro entre el código de programación y la base de datos. Su función es asegurar que cualquier dato que intente entrar al sistema cumpla con los esquemas predefinidos (tipos de datos, relaciones obligatorias), actuando como un primer filtro de seguridad contra inconsistencias.

Base de Datos

PostgreSQL. Aquí se almacena la información de manera permanente. Se escogió este motor para la base de datos relacional por su robustez en el manejo de transacciones complejas (ACID), garantizando que, en caso de un fallo eléctrico o de red durante el agendamiento de una cita, la base de datos nunca quede en un estado corrupto o incompleto.

Diagrama de Arquitectura de Software

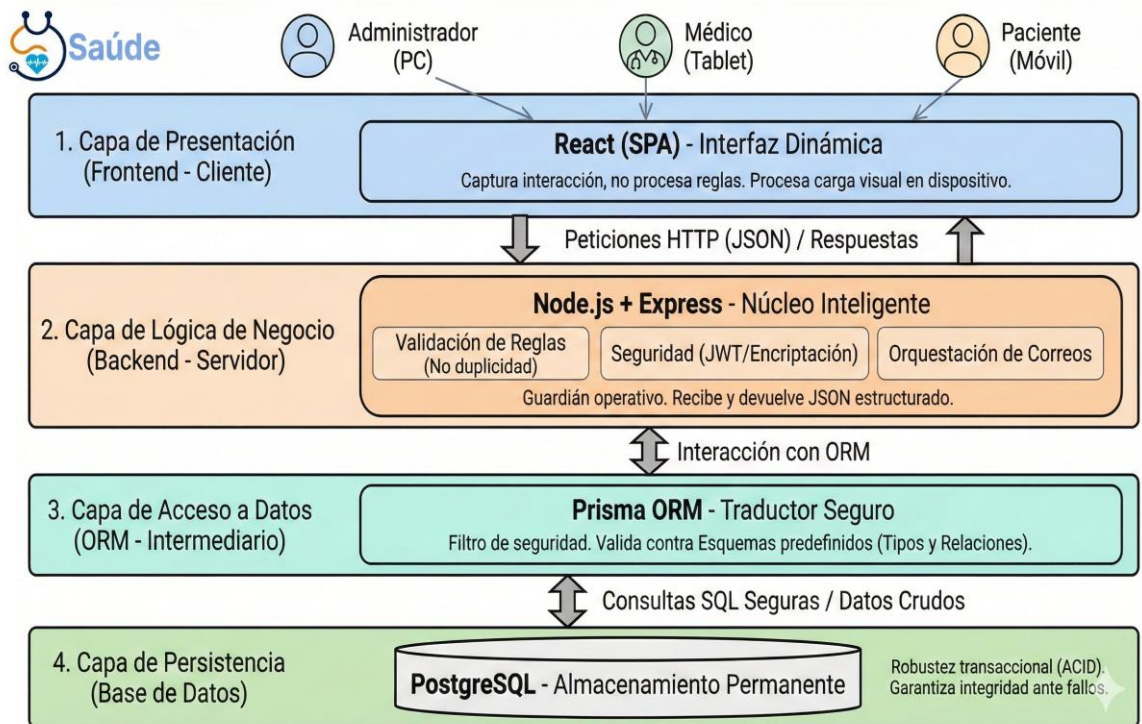
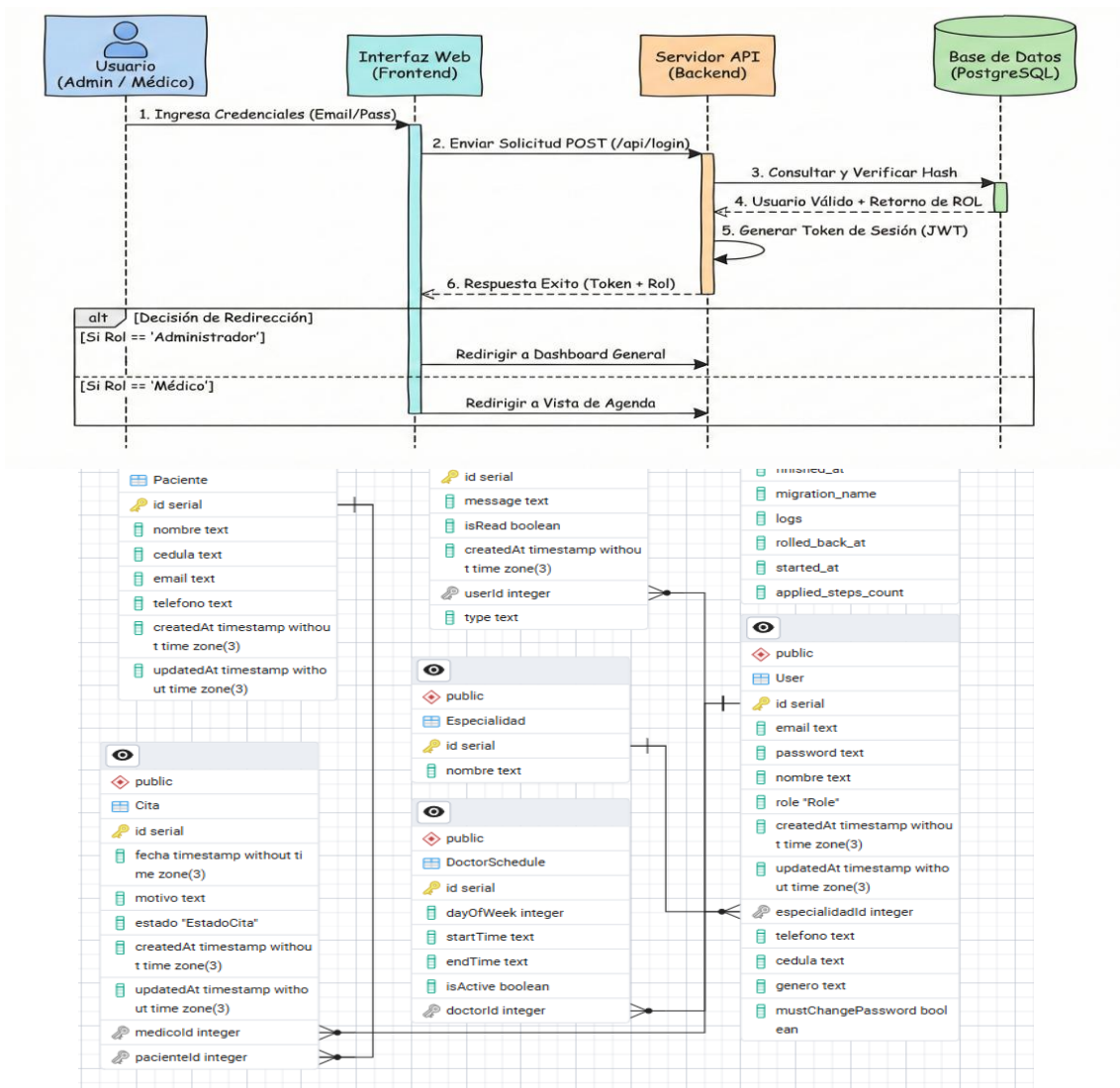


Ilustración 6

Nota: Estructura arquitectónica en capas del sistema que ilustra la segregación de funciones asignadas a la capa de presentación, el procesamiento interno y el almacenamiento de la información. El esquema representa una interfaz adaptable que facilita la interacción del usuario con la aplicación desde dispositivos móviles o tabletas.

Gráfico de Inicio de sesión de Roles, Seguridad y Base de Datos

Ilustración 7



Nota: Demuestra cómo el sistema maneja la seguridad de forma centralizada, pero deriva a los usuarios a sus áreas correspondientes de forma inteligente y el diagrama de bases de datos.

Análisis de Viabilidad Técnica

Este análisis de viabilidad técnica tiene como objetivo verificar si la infraestructura de hardware existente en el Centro Médico Saúde es suficiente para soportar la operación del nuevo sistema web. Dado que la solución ha sido diseñada bajo una arquitectura Cliente-Servidor con tecnologías web modernas (SPA), el procesamiento de la interfaz se realiza en el navegador, reduciendo significativamente la dependencia de hardware especializado o costoso delegando más el hardware requerido para la experiencia de usuario y la calidad visual.

A continuación, se detallan las especificaciones técnicas requeridas para garantizar un rendimiento fluido, estableciendo comparativas directas entre arquitecturas (Intel/AMD) por su popularidad en el mercado y justificando cada componente en función de la demanda del software.

1.1 Requisitos para Estaciones de Trabajo

Estas especificaciones aplican a los equipos utilizados por el personal administrativo y el cuerpo médico para acceder al sistema vía navegador web principalmente enfocado al uso de laptops o PC de escritorio para facilitar la experiencia de uso y navegación del usuario.

Tabla 6

Componente	Requisito Mínimo	Requisito Recomendado	Justificación Técnica
Procesador (CPU)	Intel Core i3 (6ta Gen) o AMD Ryzen 3 (1000 Series)	Intel Core i5 (8va Gen) o AMD Ryzen 5 (3000 Series)	La aplicación, al estar construida en React, ejecuta la lógica de renderizado en el navegador. Se requiere un procesador con instrucciones modernas para evitar congelamientos al cargar el calendario interactivo.
Memoria RAM	4 GB DDR3 (1600 MHz)	8 GB DDR4 (2400 MHz o superior)	Los navegadores modernos (Chrome/Brave) consumen recursos significativos por pestaña. 8 GB permiten tener abierto el sistema Saúde, el correo y archivos PDF simultáneamente sin lentitud.
Almacenamiento	Disco Duro (HDD) con 5 GB libres	Unidad de Estado Sólido (SSD)	Aunque el sistema está en la nube/servidor, el SSD acelera la carga del sistema operativo y del navegador, reduciendo el tiempo de espera al iniciar la jornada.
Pantalla	Resolución HD (1366 x 768)	Resolución Full HD (1920 x 1080)	El módulo de "Mi Agenda" despliega una grilla visual compleja. Una pantalla Full HD permite visualizar la semana completa sin necesidad de hacer scroll horizontal excesivo.
Conectividad	Ethernet 10 Mbps	Ethernet 40 Mbps o Wi-Fi 5	La comunicación con la API REST requiere una conexión estable para evitar errores al guardar citas en tiempo real.

1.2 Análisis de Despliegue

Para la puesta en producción del sistema "Saúde", se descarta la adquisición de servidores físicos debido a los altos costos iniciales de hardware, la necesidad de adecuación de un cuarto frío y la dependencia de un suministro eléctrico ininterrumpido con UPS de alto costo.

En su lugar, se opta por una infraestructura de Alojamiento en la Nube bajo el modelo IaaS (Infraestructura como Servicio) o PaaS (Plataforma como Servicio). Esta estrategia transforma la inversión de capital en un gasto operativo mensual, permitiendo al Centro Médico pagar únicamente por los recursos que consume, con flexibilidad de escalabilidad si el número de pacientes aumenta.

A continuación, se presenta una comparativa de costos proyectada para el año 2025, considerando las especificaciones técnicas requeridas (1 vCPUs + 2GB RAM) para soportar el tráfico estimado, los valores pueden ser estimaciones y pueden variar.

Tabla 7

Característica	AWS (Amazon Web Services)	Microsoft Azure
Servicio Evaluado	Amazon Lightsail (VPS Simplificado)	Azure Virtual Machines (Serie B - Burstable)
Recursos de Cómputo	1 vCPU / 2 GB RAM	1 vCPU / 1 GiB RAM (Instancia B1s)
Almacenamiento	60 GB SSD (Incluido en el precio)	64 GB Premium SSD (Se paga aparte)
Base de Datos	Alojada internamente en el servidor para reducir costos.	Alojada internamente en la VM o Azure Database for PostgreSQL (Costo extra).
Ancho de Banda	3 TB de transferencia incluidos.	Los primeros 5 GB son gratis, luego se paga por GB saliente.
Dirección IP	IP Estática incluida sin costo extra.	IP Pública Dinámica (Estática tiene costo adicional).
Costo Mensual Estimado	\$10.00 USD / mes (Precio Fijo)	\$13.50 USD / mes (Variable según disco y tráfico)
Ventaja Principal	Previsibilidad de costos. No hay sorpresas en la factura a fin de mes.	Integración nativa si el centro médico ya usa Office 365 o Windows.

CAPITULO III

Implementación

1. Alcance

El alcance del proyecto se basa en una solución web integral que trasciende el simple registro de datos. Basado en la arquitectura y los recursos definidos previamente, el sistema se construye sobre cuatro principales funcionalidades que cubren de los requerimientos operativos del centro médico “Saúde”:

1.1 Módulo de Agendamiento Inteligente

Este es el núcleo del sistema. No se limita a un formulario de registro; se implementó un motor de validación en el servidor que impide matemáticamente la superposición de horarios. El alcance cubre desde la selección visual del bloque horario por parte del recepcionista hasta la persistencia inmediata en la base de datos, garantizando que un médico nunca tenga dos pacientes citados al mismo tiempo.

1.2 Sistema de Notificaciones y Recordatorios

Para abordar el problema del ausentismo, el alcance incluye la automatización de las comunicaciones. Se integró un servicio de correo (SMTP) que actúa en dos momentos clave:

- **Confirmación Inmediata:** Disparada al momento de crear la cita.
- **Recordatorio Preventivo:** Ejecutado por una tarea programada que revisa la agenda diariamente y alerta a los pacientes 24 horas antes de su consulta, sin intervención humana.

1.3 Administración de Pacientes y Especialidades

Se desarrolló una interfaz de gestión que permite crear el periodo de vida completo de la información (Crear, Leer, Actualizar). Además, se habilitó un componente público (Landing Page) para mostrar a los visitantes qué especialidades están activas, funcionando como una vitrina digital del centro médico que se mantiene actualizada automáticamente desde el panel administrativo.

1.4 Segregación de Paneles (Administrador, Médico)

El proyecto cumplió con la necesidad de seguridad y privacidad mediante la creación de entornos separados:

- **Panel Administrativo:** Con control total sobre la configuración global, usuarios y auditoría de todas las agendas.
- **Panel Médico:** Una vista restringida y optimizada donde el especialista solo accede a su propia agenda y al historial de sus pacientes, protegiendo la confidencialidad de la información clínica ajena a su servicio.

2. Arquitectura Cliente – Servidor

2.1 Arquitectura y Flujos de Comunicación

La implementación, se estructura el sistema "Saúde" bajo un modelo de arquitectura **Cliente-Servidor**. Esta decisión transforma radicalmente la operativa del centro médico: en lugar de depender de archivos almacenados en una sola computadora, como sucedía (TO-BE) con las hojas de cálculo, ahora existe una separación clara entre la interfaz que usan las personas y el cerebro que procesa los datos.

En términos prácticos, la implementación se divide en dos grandes bloques que se comunican constantemente:

a. El Cliente (La cara visible):

Desarrollado para ser accesible desde cualquier navegador. Aquí es donde los médicos y recepcionistas ven las pantallas, los calendarios y los formularios. Su única función es mostrar la información de forma amigable y capturar lo que el usuario desea hacer (iniciar sesión, agregar, agendar, cancelar).

b. El Servidor y la API REST:

Es el motor que trabaja en la nube. Su trabajo es recibir las órdenes del cliente, verificar que quien las pide tenga permiso (seguridad) y guardar o buscar la información en la base de datos. Para que el Cliente y el Servidor se entiendan, se implementó una API REST. Esta funciona como un "puente": el Cliente envía una solicitud (por ejemplo, "Guardar Cita") y la API se encarga de procesarla y devolver una respuesta de éxito o error.

2.2 Organización Modular del Sistema

Para evitar que el software sea un caos de código mezclado, la implementación se organizó en módulos independientes, replicando las áreas funcionales del centro médico:

- **Módulo Público:**

Es la puerta de entrada. Permite a cualquier persona ver los servicios y especialidades sin necesidad de registros innecesarios, esto ayuda a la experiencia de usuario, ya que el posible cliente quiere saber si en el centro se ofrece el servicio que busca y no desea gestionar una contraseña más, teniendo en cuenta que una gran parte pueden ser adultos mayores con poco manejo de la tecnología. Por otra parte, el personal se dirige hacia el inicio de sesión.

- **Módulo Administrativo:**

Es el centro de control. Desde aquí se configuran los horarios globales, se crean las cuentas de los doctores y se tiene una visión completa de la operatividad del negocio.

- **Módulo Médico:**

Diseñado exclusivamente para la atención clínica. Aquí el doctor tiene su agenda personal digitalizada y el acceso directo al historial de sus pacientes, sin distracciones ni funciones administrativas que no le corresponden.

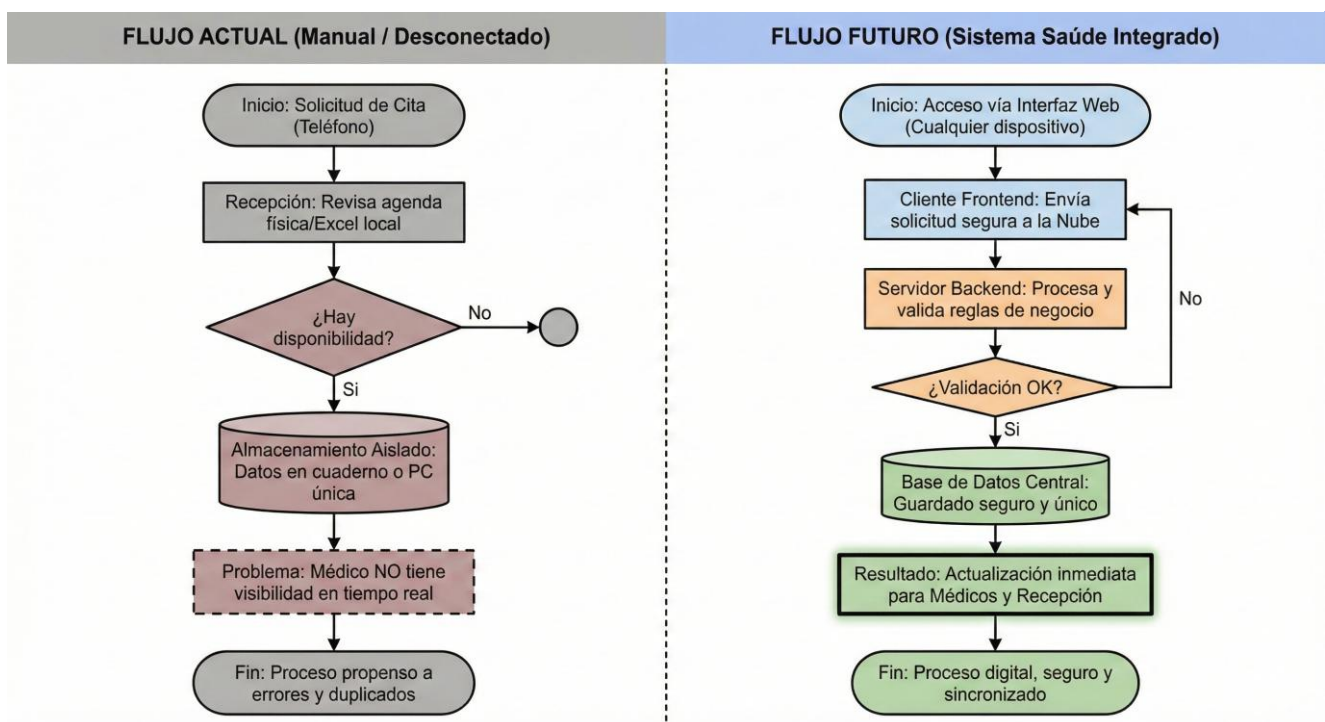
Flujo Básico de Comunicación

Con esta implementación, los procesos manuales se reemplazan por flujos de datos automatizados. El recorrido de la información sigue siempre este camino lógico:

- a. **Interacción:** El usuario realiza una acción en la pantalla (ej. clic en "Confirmar Cita").

- b. **Solicitud:** El sistema envía esos datos al servidor de forma segura e invisible para el usuario.
- c. **Procesamiento:** El servidor recibe la orden, revisa que el horario esté libre (validación de negocio) y guarda la información.
- d. **Respuesta:** El servidor avisa que "todo salió bien" y la pantalla del usuario se actualiza instantáneamente mostrando la cita agendada en el calendario.

Ilustración 8



Nota: Diagrama de flujo comparativo. A la izquierda, el proceso manual actual con almacenamiento de datos aislado. A la derecha, el flujo futuro integrado mediante la arquitectura Cliente-Servidor y base de datos

3. ESTÁNDAREZ DE INTERFAZ Y USABILIDAD

3.1 Estándares de Uso y Experiencia de Usuario (UX)

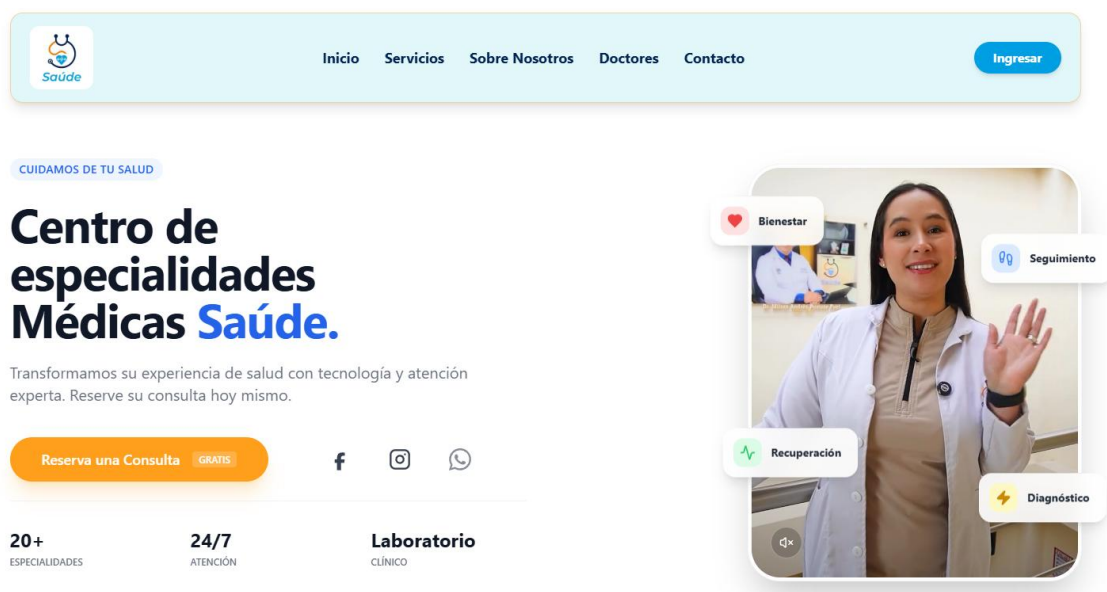
El diseño de la interfaz de "Saúde" no se basó únicamente en criterios estéticos, sino en la necesidad funcional de reducir la carga cognitiva del personal médico y administrativo. Dado que el sistema se utiliza en un entorno de atención al paciente, donde la velocidad y la precisión son críticas, se aplicaron estándares internacionales para garantizar que la interacción sea intuitiva y libre de errores.

A continuación, se detalla la implementación de estos principios en las pantallas del sistema:

3.2 Accesibilidad Web (WCAG AA) y Principios POUR

Para asegurar que la plataforma sea inclusiva y operable en diversas condiciones, se ajusta colores pastel basada en la paleta de colores del centro médico Saúde. Estéticamente es suave, pero se asegura la distinción entre el texto y fondo para cumplir con el mínimo requerido para un nivel AA de las pautas accesibilidad a la web (WCAG).

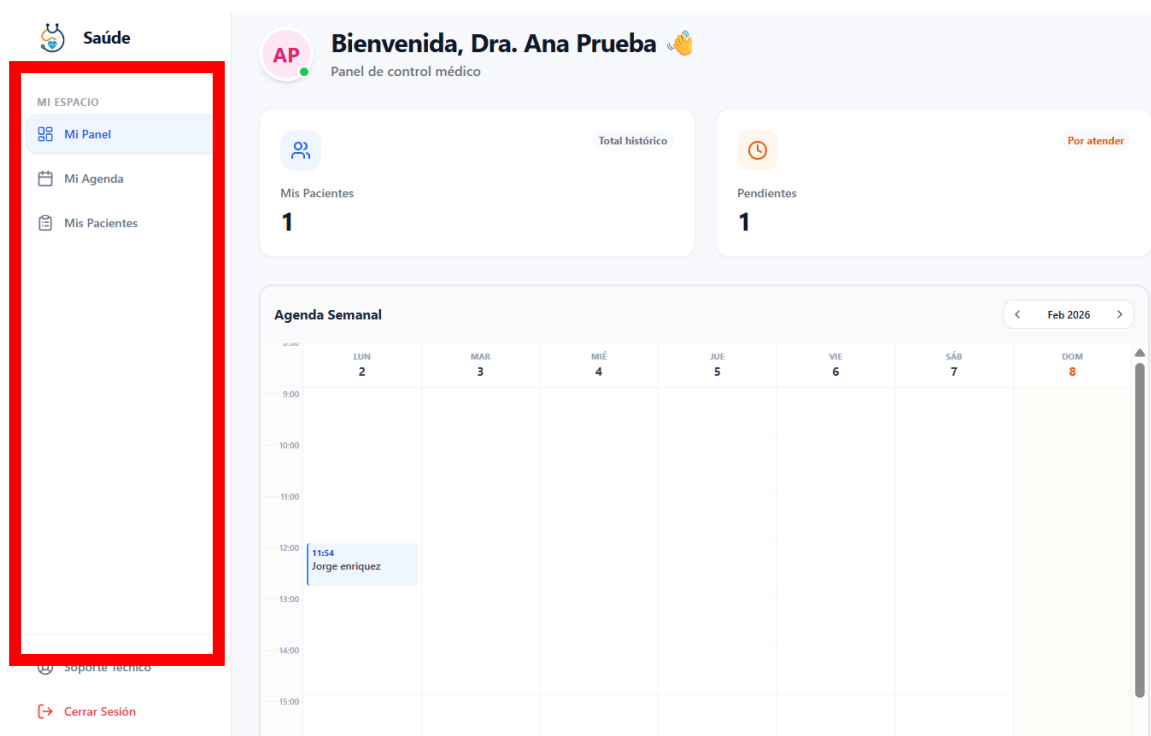
Ilustración 9



Bajo los principios POUR, se realizaron las siguientes adecuaciones:

- a. Perceptible: Todos los iconos del menú (Agenda, Pacientes, Salir) están acompañados de etiquetas de texto, evitando la ambigüedad.

Ilustración 10



- b. Operable: Los formularios de registro de pacientes son completamente navegables mediante el teclado (Tabulador), permitiendo a la recepción ingresar datos sin depender del mouse.
- c. Comprensible: Los mensajes de error no muestran códigos técnicos (ej. "Error 500"), sino instrucciones claras en lenguaje natural, como completar un campo necesario que se deja vacío.

Ilustración 11

The image shows a web form titled "Nuevo Paciente" with a close button (X) in the top right corner. The form contains three input fields: "NOMBRE" (Name) with the value "Jorge Yopez", "CÉDULA" (ID Card), and "EMAIL". The "EMAIL" field is highlighted with a red oval and contains the text "jorge@gmail.com" followed by a masked section "0999999999". A validation error message, "Completa este campo" (Complete this field), is displayed above the "EMAIL" field, accompanied by an orange exclamation mark icon. At the bottom of the form, there are two buttons: "Cancelar" (Cancel) and "Registrar" (Register).

Nota: Muestra un error preventivo en caso de omitir un campo, todos son obligatorios y con formato.

3.3 Aplicación de las Heurísticas de Nielsen

La usabilidad del sistema se auditó utilizando los principios de Jakob Nielsen y se destaca:

1. Visibilidad de estado en el sistema:

El usuario siempre sabe que está pasando (carga, guardador, error) (ilustración 10).

2. Prevención de Errores:

Los formularios validan los datos ingresados con formatos establecidos para evitar en ingreso de datos erróneos (ilustración 11).

3.4 Diseño Centrado en la Experiencia de usuario

Un sistema de navegación limpio, para completar que una tarea requiera simplicidad y se haga con pocos clics. Un caso de ejemplo es el usuario médico al ingresar mantiene la vista de su calendario destacando el ‘hoy’ para visualizar sus citas del día y también las de los siguientes días sin necesidad filtros de fechas (Figura 10).

4. GESTIÓN DE PACIENTES

El apartado de gestión de pacientes permite que el personal ingrese la información del paciente, asigne un médico, editar sus datos o eliminar al paciente del sistema. Los formatos establecidos en el backend validan datos como la cédula de identidad, dirección de correo electrónico o número telefónico, facilitando el ingreso y posteriormente acceso a la información del paciente

The illustration shows a user interface for patient management. At the top, there are two modal forms: 'Nuevo Paciente' (New Patient) and 'Editar Paciente' (Edit Patient). Both forms have fields for 'NOMBRE' (Name), 'CÉDULA' (ID Card), 'EMAIL', and 'TELÉFONO' (Phone Number). The 'Nuevo Paciente' form has a 'Registrar' button, while the 'Editar Paciente' form has a 'Guardar' button. Below these forms is a main section titled 'Pacientes' with the subtitle 'Directorio general de pacientes registrados'. It includes a search bar labeled 'Buscar...' and a '+ Nuevo Paciente' button. Below the search bar is a table with the following columns: 'NOMBRE DEL PACIENTE', 'CONTACTO', 'CÉDULA', 'DOCTOR ASIGNADO', and 'ACCIONES'. The table contains one entry for 'Jorge enriquez' with ID 'P-1003'. The 'ACCIONES' column for this entry shows a dropdown menu with 'Editar' and 'Eliminar' options. The 'DOCTOR ASIGNADO' column shows 'Ana Mai Pru Urok'.

NOMBRE DEL PACIENTE	CONTACTO	CÉDULA	DOCTOR ASIGNADO	ACCIONES
 Jorge enriquez ID: P-1003	 9000000000  xs@saude.com	1234567890	Ana Mai Pru Urok	 Editar  Eliminar

Ilustración 12

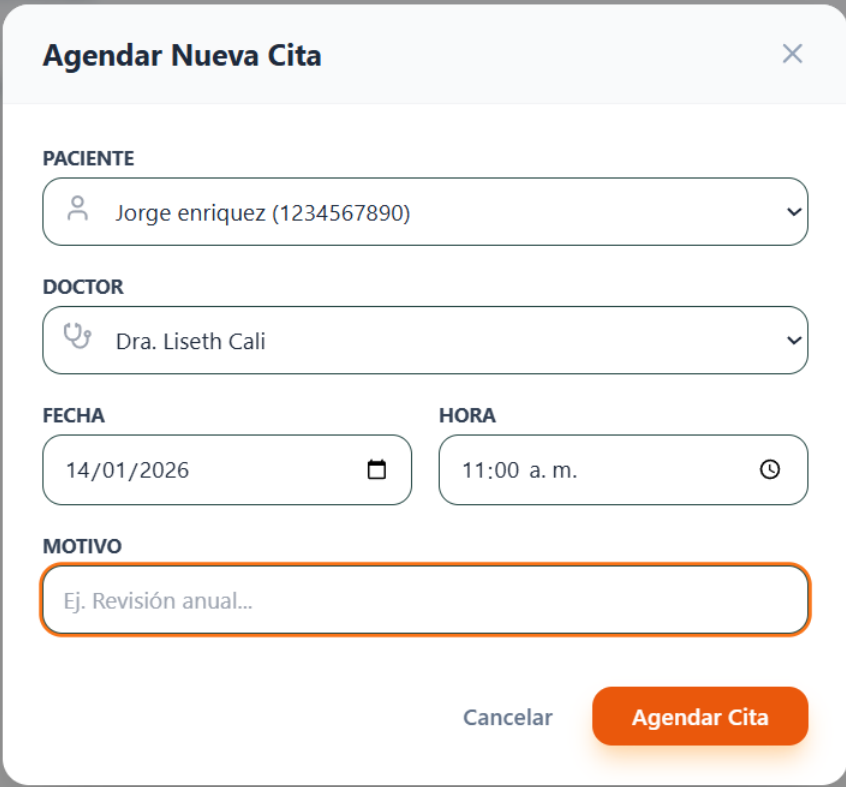
Nota: Indica añadir, editar, agendar o eliminar a un paciente.

5. Módulos funcionales

Aquí es donde el sistema nuevo para Saúde optimiza el trabajo del equipo. La idea no es simplemente pasar los datos a una computadora, sino diseñar herramientas para eliminar las tareas repetitivas del día a día y retroalimentar en caso de errores.

5.1 Módulos para Gestión de Citas

En lugar de recurrir al uso hojas físicas, el recepcionista simplemente registra los datos del paciente en el sistema y agenda una nueva cita en el botón “agendar cita”. El sistema despliega un formulario, que permite asignar al médico si esta disponible y agregar el horario, fecha de la cita.



Agendar Nueva Cita ✕

PACIENTE

👤 Jorge Enriquez (1234567890) ▾

DOCTOR

👤 Dra. Liseth Cali ▾

FECHA **HORA**

14/01/2026 📅 11:00 a. m. ⌚

MOTIVO

Ej. Revisión anual...

Cancelar **Agendar Cita**

Ilustración 13

5.2 Catálogo de Especialidades Médicas

Se construyó un módulo que funciona como un catálogo digital de la oferta de especialidades que tiene el centro médico, se visualiza en la segunda sección de la página principal.

Ilustración 14



Nota: Oferta de Especialidades médicas en página principal.

5.3 Panel Administrativo Centralizado

Es el área de trabajo del administrador. Desde aquí se gestionan las tareas principales, como crear de nuevas cuentas de usuario y la definir horarios laborales en caso de los médicos y añadir nuevos pacientes y gestionar las citas en caso de los pacientes.

Ilustración 15

Registrar Nuevo Doctor

NOMBRE COMPLETO
Juliana Lucía Espinel Cortez

CÉDULA
0179874848

GÉNERO
Femenino

EMAIL
jlespinel@saude.com

TELÉFONO
09000087654

ESPECIALIDAD
Seleccione...

Cancelar Crear

Lunes 09:00 a. m. a 05:00 p. m. ✓

Martes 09:00 a. m. a 05:00 p. m. ✓

Miércoles 09:00 a. m. a 05:00 p. m. ✓

Jueves 09:00 a. m. a 05:00 p. m. ✓

Viernes 09:00 a. m. a 05:00 p. m. ✓

Sábado 09:00 a. m. a 05:00 p. m.

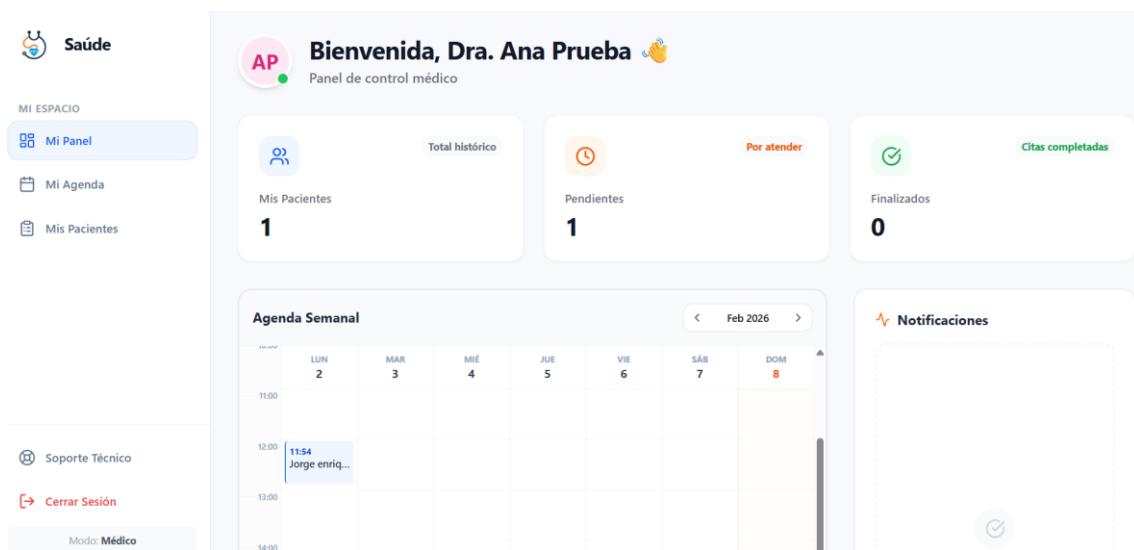
Domingo 09:00 a. m. a 05:00 p. m.

Cancelar Guardar Horario

5.4 Panel Médico

Este módulo fue diseñado para tener entorno de trabajo focalizado, eliminando cualquier vista innecesaria de tareas administrativas, así el personal médico centra su visualización exclusivamente en sus actividades diarias. A diferencia del panel administrativo, esta interfaz presenta una vista filtrada y simplificada de la información.

Ilustración 16



5.5 Optimización del Tiempo Operativo

La integración de estos módulos permite un flujo de trabajo lineal y sin interrupciones: el Administrador configura la disponibilidad, Módulo de Citas, tiene acceso a las especialidades y pacientes. El médico tiene acceso solo a su agenda en forma de calendario y a sus pacientes asignados. Esta arquitectura reduce el tiempo total de agendamiento de minutos (en el proceso manual/telefónico) a segundos, minimizando el error humano y mejorando la percepción de eficiencia del servicio.

6. Recordatorios

6.1 Automatización de Notificaciones y Recordatorios

Para mitigar el problema del ausentismo en las consultas, se implementó un servicio de mensajería automatizado que opera de forma independiente a la intervención humana. La solución técnica se construyó utilizando Nodemailer como cliente de transporte SMTP dentro del servidor Node.js, configurado para enviar correos transaccionales con plantillas personalizadas del centro, datos de la cita y ubicación.

6.2 La lógica de notificación:

a. Confirmación Inmediata

Funciona en tiempo real. En el momento exacto en que la recepcionista o el paciente finalizan el agendamiento, el controlador del backend dispara una petición al servicio de correo. El sistema no confirma la cita en pantalla hasta que el correo de respaldo ha sido enviado exitosamente, garantizando que el paciente tenga el comprobante digital en su bandeja de entrada al instante.

b. Recordatorio Preventivo de 24 Horas

Para el aviso previo, se implementó una tarea en segundo plano utilizando la librería node-cron. Este proceso es invisible para los usuarios y funciona de la siguiente manera:

- **Ejecución Automática:** El servidor ejecuta una rutina de revisión todos los días a las 08:00 AM.
- **Consulta Inteligente:** El script conecta con la base de datos PostgreSQL y busca todas las citas cuya fecha programada sea "mañana".

- **Envío Masivo:** Una vez identificados los pacientes, el sistema envía los recordatorios de forma secuencial.

Esta implementación libera al personal de recepción de la tarea manual de llamar uno por uno a los pacientes del día siguiente, reduciendo la carga operativa y asegurando que ningún paciente olvide su compromiso médico.

7. Seguridad

7.1 Implementación de Seguridad y Control de Acceso

Dada la naturaleza sensible de la información médica que maneja "Saúde", la seguridad no fue tratada como un módulo opcional, sino como una capa transversal en toda la arquitectura. Se implementaron mecanismos de protección tanto en el tránsito de los datos como en la validación de la identidad de los usuarios.

7.2 Autenticación y Gestión de Sesiones (JWT)

Para el inicio de sesión, se abandonó el uso de cookies de sesión tradicionales en favor de JSON Web Tokens (JWT). Este estándar permite una autenticación sin estado, donde el servidor no necesita almacenar la sesión en memoria. Funcionamiento: Cuando el usuario ingresa sus credenciales de forma correcta, el backend firma digitalmente un token encriptado que contiene su ID y su Rol. Este token se almacena de forma segura en el navegador del cliente y debe ser enviado en cada petición subsiguiente. Se aplicó un esquema de Control de Acceso Basado en Roles (RBAC) para garantizar que cada usuario acceda únicamente a las funciones pertinentes a su cargo.

- a. **Administrativo:** Posee permisos de escritura y lectura global sobre la configuración del sistema y el catálogo de usuarios.
- b. **Médico:** Su alcance está restringido por middleware en el servidor. Aunque un médico intente acceder a la URL del panel administrativo manualmente, el sistema verifica el rol en el token JWT y bloquea el acceso, redirigiéndolo a su agenda personal.

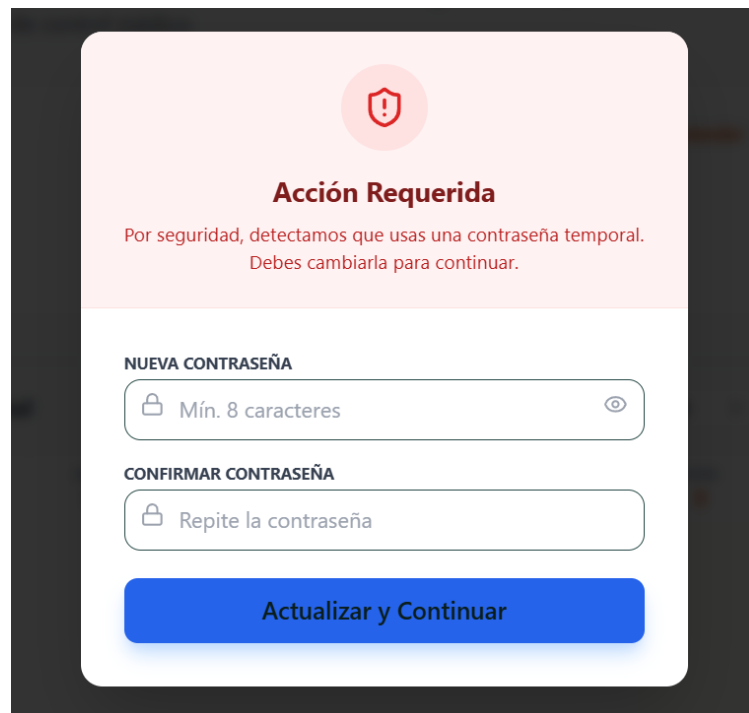
7.3 Despliegue Seguro (HTTPS)

El sistema en producción opera exclusivamente bajo el protocolo HTTPS. SSL/TLS que cifra toda la comunicación entre el servidor y el navegador del usuario en la nube.

7.4 Política de Primer Acceso y Cambio de Clave

Como medida de seguridad adicional para la gestión de identidades, se diseñó un flujo específico para el alta de nuevos médicos:

- a. Creación: El administrador genera la cuenta del médico con una contraseña temporal que es su número de identificación.
 - b. Detección: Al iniciar sesión por primera vez el nuevo usuario médico, el sistema detecta que la cuenta tiene el estado de "Activación Requerida".
 - c. Bloqueo y Cambio Forzoso: La interfaz bloquea el acceso al menú principal y redirige obligatoriamente al usuario a una pantalla de "Cambio de Contraseña".
- El médico no puede operar el sistema hasta que defina una nueva clave personal que cumpla con los requisitos de longitud y complejidad, garantizando que solo él conozca sus credenciales de acceso.



Acción Requerida

Por seguridad, detectamos que usas una contraseña temporal.
Debes cambiarla para continuar.

NUEVA CONTRASEÑA

Mín. 8 caracteres

CONFIRMAR CONTRASEÑA

Repite la contraseña

Actualizar y Continuar

Ilustración 17

Nota: La viñeta muestra el sistema de actualización de contraseña que el sistema pide una vez que un nuevo personal médico accede por primera vez.

CONCLUSIONES

-La implementación de la arquitectura Cliente-Servidor mediante APIREST centralizó exitosamente la operativa del centro médico, eliminando la dependencia de archivos locales aislados haciendo posible la sincronización de datos en tiempo real entre la recepción y los consultorios.

-El diseño de la interfaz, alineado con estándares de usabilidad y accesibilidad, reduce significativamente la curva de aprendizaje del personal, permite que recepcionistas y médicos adopten el sistema intuitivamente sin necesidad de capacitación técnica extensiva ni manuales complejos.

-En el panel de administración de pacientes se garantizó el correcto funcionamiento de la base de datos a través de la aplicación de reglas de formato para la información en el servidor, evitando la creación de registros duplicados y garantizando que la información para el registro de datos.

-Se integró los diferentes módulos, cambiando un proceso manual desordenado a uno digital unificado que disminuye drásticamente el tiempo operativo para agendar, modificar y consultar la disponibilidad de los doctores.

-Los requisitos de protección de datos se cumplieron a través de JWT y la segregación rígida de roles.

- El sistema para notificar el agendamiento de una cita y el recordatorio preventivos de 24 horas reemplaza la tarea manual de llamadas para

confirmación, optimizando el tiempo de la recepción y entregando al paciente un recordatorio digital inmediato de su compromiso médico.

-El análisis de recursos confirmó que la solución es totalmente funcional en infraestructuras de nube de bajo costo, demostrando que la modernización del centro médico es financieramente sostenible y no requiere inversiones en hardware físico especializado.

RECOMENDACIONES

-Implementación de un portal de gestión para Pacientes. Se recomienda extender la arquitectura actual para incluir un tercer rol de usuario: el "Paciente". Esto permitiría que los usuarios pacientes que mantengan citas de seguimiento o tratamientos puedan visualizar las notas y recomendaciones de su médico en su panel, así como sus citas programadas e historial de las mismas.

-Integración con API de WhatsApp Business. Aunque el sistema de correos funciona, se sugiere evolucionar el módulo de notificaciones integrando el api de whatsapp. Dado el contexto cultural actual de la inmediatez de la mensajería instantánea, esto aumentaría drásticamente la tasa de lectura de los recordatorios, reduciendo el ausentismo de pacientes que no revisan su correo electrónico con frecuencia.

-Reportes Avanzados, esto permitiría a la dirección visualizar métricas clave como "horarios de mayor demanda", "especialidades más rentables" o "tasa de cancelación por médico", transformando los datos acumulados en información

estratégica de marketing e información contable para la tomar decisiones empresariales y contratación de personal.

REFERENCIAS BIBLIOGRÁFICAS

- Git. (s.f.). *git-scm.com*. Obtenido de <https://git-scm.com/docs/git>
- GitHub. (s.f.). *Documentación de GitHub: Acerca de GitHub y Git*. Obtenido de <https://docs.github.com/es/get-started>
- Organización Panamericana de la Salud. (2021). *Ocho principios rectores de la transformación digital del sector de la salud*. Washington, D.C.: OPS.
- sandrockj. (17 de November de 2024). *El ORM Prisma: una breve descripción general e introducción*. Obtenido de <https://dev.to/sandrockjustin/the-prisma-orm-a-brief-overview-and-introduction-353m>
- Schwaber, K., & Sutherland, J. (2020). *scrumguides.org*. Obtenido de [scrumguides.org: https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf#zoom=100](https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf#zoom=100)
- The PostgreSQL Global Development Group. (13 de November de 2025). *What Is PostgreSQL?* Obtenido de www.postgresql.org: <https://www.postgresql.org/docs/current/intro-what-is.html>

ANEXOS

LISTA DE TABLAS

Tabla 1

Componente	Elemento	Aplicación en el Proyecto
Enfoque de Investigación	Cualitativo	Diálogos y observación directa para levantar necesidades y definir criterios de aceptación.
	Cuantitativo	Establecimiento de línea base (tiempos y errores) para contrastar resultados post-implementación.
Marco de Trabajo (Scrum)	Sprints	Iteraciones de duración fija que funcionan como el "ritmo cardíaco" para entregar valor continuo.
	Planificación	Selección de tareas desde el Product Backlog (lista ordenada de necesidades).
	Ejecución	Transformación de requerimientos en incrementos de software funcional.
	Revisión	Inspección del funcionamiento real para ajustar el plan y asegurar transparencia.
Herramientas de Apoyo	Git	Sistema de control de versiones distribuido para gestionar el historial de cambios.
	GitHub	Plataforma en la nube para alojamiento de código y gestión de tareas mediante <i>Issues</i> .

Tabla 2

Tarea	Responsable	Duración
Reunión de inicio con la dirección médica para definir el alcance del proyecto.	Andrés Sierra	2 días
Observación en sitio del proceso manual de agendamiento y atención al paciente.	Andrés Sierra	3 días
Inspección técnica de los equipos de cómputo e internet del centro médico.	Andrés Sierra	1 día
Redacción y validación de la lista de requerimientos del sistema.	Andrés Sierra	2 días
Diseño del esquema de la base de datos (tablas de médicos, pacientes y citas).	Andrés Sierra	3 días
Preparación del entorno de desarrollo (local).	Andrés Sierra	2 días
Creación del repositorio de código y estructura de carpetas del proyecto.	Andrés Sierra	1 día

Ajuste inicial de la base de datos y enlace con el servidor.	Andrés Sierra	1 día
--	---------------	-------

Tabla 3

Sprint II: Desarrollo del Frontend y Experiencia de Usuario	Duración: 3 Semanas	
Tarea	Responsable	Duración
Configuración inicial del proyecto visual y definición de la paleta de colores del centro médico.	Andrés Sierra	2 días
Diseño y programación de componentes reutilizables (Botones, campos de texto, tarjetas de información).	Andrés Sierra	2 días
Desarrollo de la pantalla de inicio de sesión (Login) e integración con la seguridad del servidor.	Andrés Sierra	2 días
Construcción del menú de navegación lateral.	Andrés Sierra	2 días
Desarrollo de la interfaz para el listado de pacientes.	Andrés Sierra	2 días
Implementación de los formularios de registro y edición de historias clínicas.	Andrés Sierra	2 días
Diseño de la vista principal del calendario para la gestión visual de citas.	Andrés Sierra	3 días
Conexión de las pantallas del módulo administrativo con la base de datos.	Andrés Sierra	2 días
Desarrollo visual Calendario de citas.	Andrés Sierra	1 día
Conexión de rutas entre panel Administrativo y médico.	Andrés Sierra	1 día

Tabla 4

Sprint III: Funcionalidades Críticas y Gestión de Agenda	Duración: 3 Semanas	
Tarea	Responsable	Duración
Desarrollo del endpoint y lógica para detectar el primer inicio de sesión del usuario.	Andrés Sierra	1 día
Implementación del flujo de cambio de contraseña obligatorio por seguridad en el primer acceso.	Andrés Sierra	2 días
Programación del algoritmo de validación en el servidor para impedir choques de horarios entre citas.	Andrés Sierra	3 días
Integración y personalización de la librería de calendario interactivo en el Frontend.	Andrés Sierra	2 días
Desarrollo de la vista "Mi Agenda" filtrada exclusivamente para el usuario con rol de Médico.	Andrés Sierra	3 días
Programación de la interactividad del calendario: selección de bloques horarios y modales de detalle.	Andrés Sierra	3 días
Conexión de la interfaz de agendamiento con la base de datos (Carga y guardado de citas).	Andrés Sierra	2 días
Implementación de la funcionalidad para cancelar o reprogramar citas desde el calendario.	Andrés Sierra	2 días
Ajustes visuales y de usabilidad en la visualización de la agenda por día, semana y mes.	Andrés Sierra	1 día

Tabla 5

Sprint IV: Notificaciones, Módulo Público y Entrega Final	Duración: 3 Semanas	
Tarea	Responsable	Duración
Configuración del servicio de envío de correos (usando cuenta Gmail y SMTP) en el servidor.	Andrés Sierra	2 días
Programación de la notificación automática al paciente inmediatamente después de agendar una cita.	Andrés Sierra	2 días

Implementación de la tarea automática diaria (6:00 AM) para enviar recordatorios de citas de las próximas 24 horas.	Andrés Sierra	3 días
Mejora de la sección "Especialidades" en la página principal (Landing Page) para información pública.	Andrés Sierra	3 días
Pruebas integrales del sistema: validar que los correos electrónicos lleguen a la bandeja de entrada en lugar de la bandeja spam.	Andrés Sierra	2 días
Corrección de detalles estéticos finales y errores menores reportados en las pruebas.	Andrés Sierra	2 días
Preparación del entorno final de despliegue.	Andrés Sierra	1 día

Tabla 6

Componente	Requisito Mínimo	Requisito Recomendado	Justificación Técnica
Procesador (CPU)	Intel Core i3 (6ta Gen) o AMD Ryzen 3 (1000 Series)	Intel Core i5 (8va Gen) o AMD Ryzen 5 (3000 Series)	La aplicación, al estar construida en React, ejecuta la lógica de renderizado en el navegador. Se requiere un procesador con instrucciones modernas para evitar congelamientos al cargar el calendario interactivo.
Memoria RAM	4 GB DDR3 (1600 MHz)	8 GB DDR4 (2400 MHz o superior)	Los navegadores modernos (Chrome/Brave) consumen recursos significativos por pestaña. 8 GB permiten tener abierto el sistema Saúde, el correo y archivos PDF simultáneamente sin lentitud.
Almacenamiento	Disco Duro (HDD) con 5 GB libres	Unidad de Estado Sólido (SSD)	Aunque el sistema está en la nube/servidor, el SSD acelera la carga del sistema operativo y del navegador, reduciendo el tiempo de espera al iniciar la jornada.
Pantalla	Resolución HD (1366 x 768)	Resolución Full HD (1920 x 1080)	El módulo de "Mi Agenda" despliega una grilla visual compleja. Una pantalla Full HD permite visualizar la

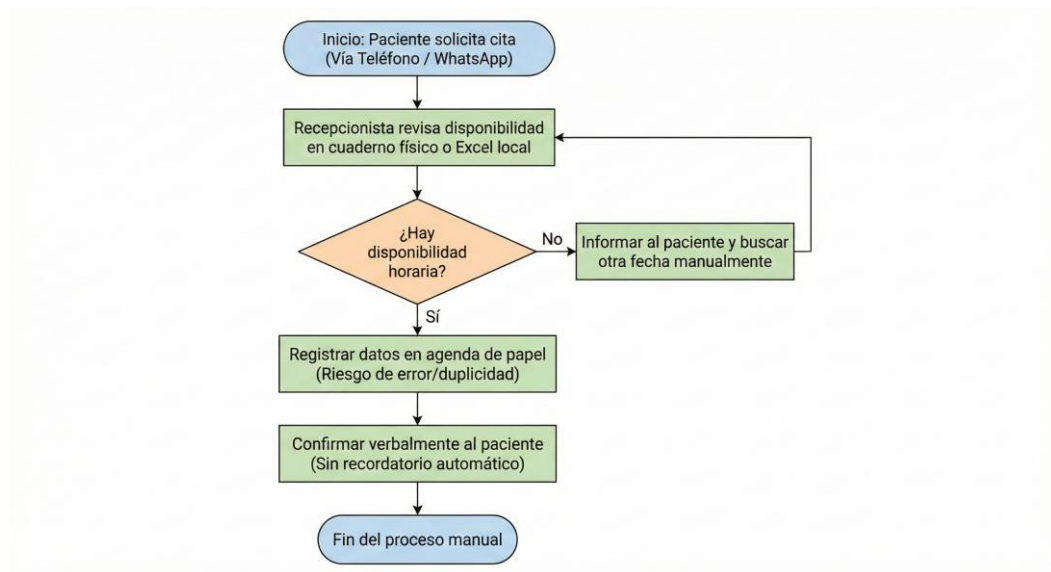
			semana completa sin necesidad de hacer scroll horizontal excesivo.
Conectividad	Ethernet 10 Mbps	Ethernet 40 Mbps o Wi-Fi 5	La comunicación con la API REST requiere una conexión estable para evitar errores al guardar citas en tiempo real.

Tabla 7

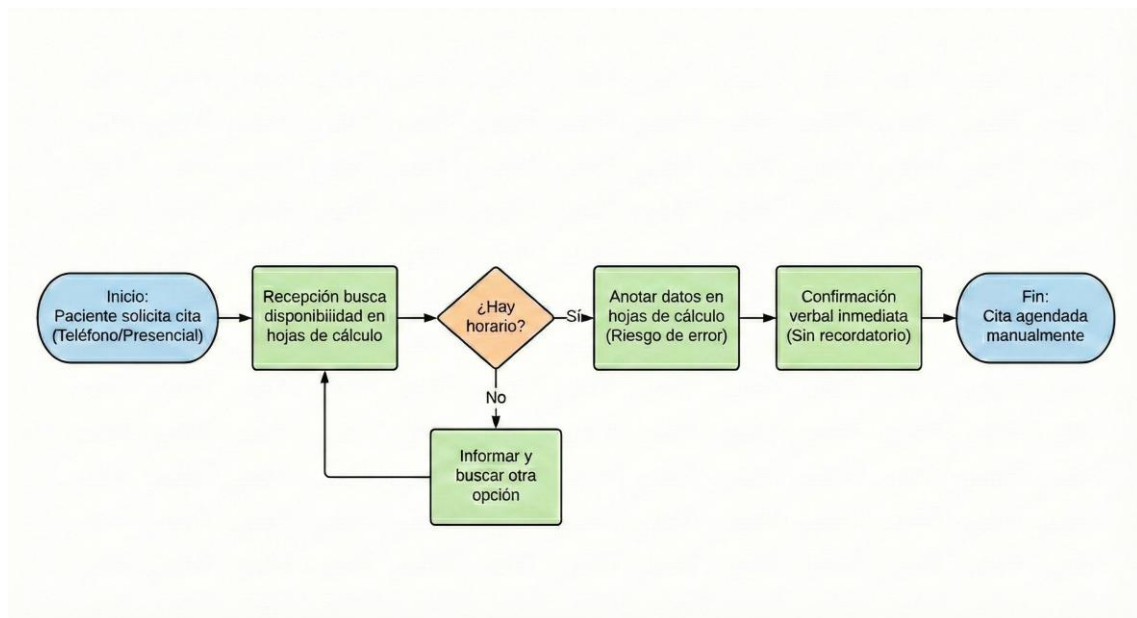
Característica	AWS (Amazon Web Services)	Microsoft Azure
Servicio Evaluado	Amazon Lightsail (VPS Simplificado)	Azure Virtual Machines (Serie B - Burstable)
Recursos de Cómputo	1 vCPU / 2 GB RAM	1 vCPU / 1 GiB RAM (Instancia B1s)
Almacenamiento	60 GB SSD (Incluido en el precio)	64 GB Premium SSD (Se paga aparte)
Base de Datos	Alojada internamente en el servidor (Dockerizada) para reducir costos.	Alojada internamente en la VM o Azure Database for PostgreSQL (Costo extra).
Ancho de Banda	3 TB de transferencia incluidos.	Los primeros 5 GB son gratis, luego se paga por GB saliente.
Dirección IP	IP Estática incluida sin costo extra.	IP Pública Dinámica (Estática tiene costo adicional).
Costo Mensual Estimado	\$10.00 USD / mes (Precio Fijo)	~\$13.50 USD / mes (Variable según disco y tráfico)
Ventaja Principal	Previsibilidad de costos. No hay sorpresas en la factura a fin de mes.	Integración nativa si el centro médico ya usa Office 365 o Windows.

LISTA DE FIGURAS

Figura 1. Proceso de actual de operaciones administrativas

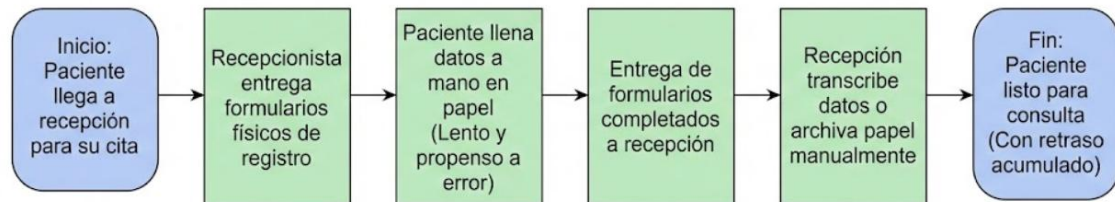


- Nota. La figura del diagrama de flujo muestra los múltiples pasos manuales requeridos para confirmar una sola cita.
- **Figura 2.** Proceso manual de agendamiento de citas en Saúde



Nota. El diagrama ilustra el flujo actual donde la disponibilidad se verifica en hojas de cálculo, generando riesgo de duplicidad de horarios.

- **Figura 3.** Proceso manual de registro y admisión de pacientes



Nota. Representación del llenado de formularios en papel previo a la consulta, lo que retrasa el tiempo de atención efectiva.

Figura 4

Nota: Diagrama Navegación general que ilustra la segregación de interfaces y permisos de acceso para los roles de administrador y médico tras la autenticación.

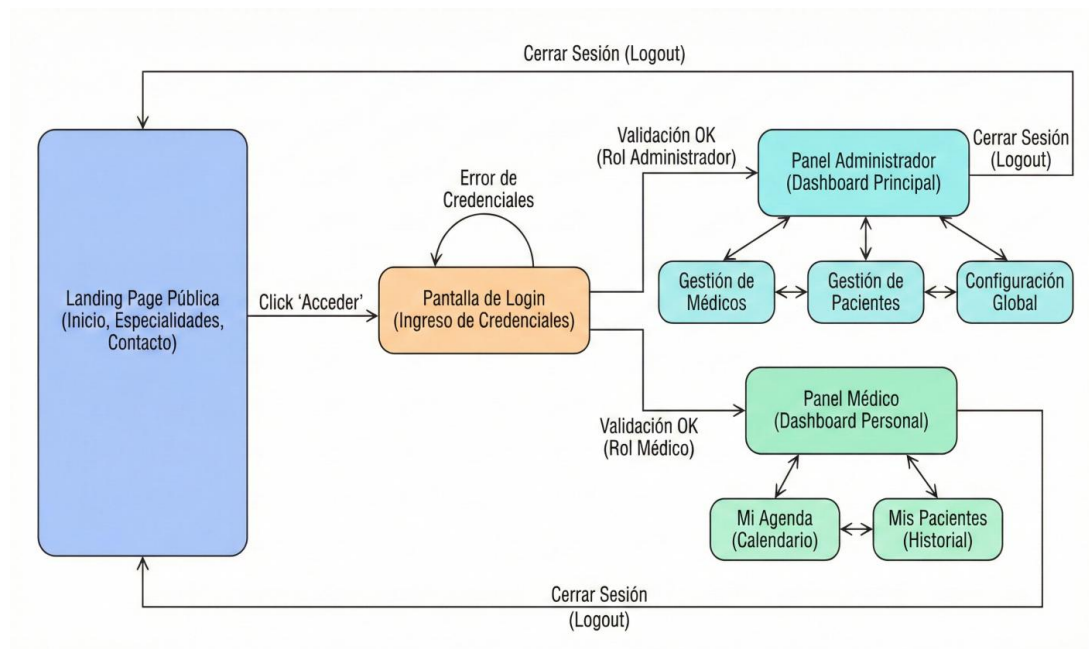
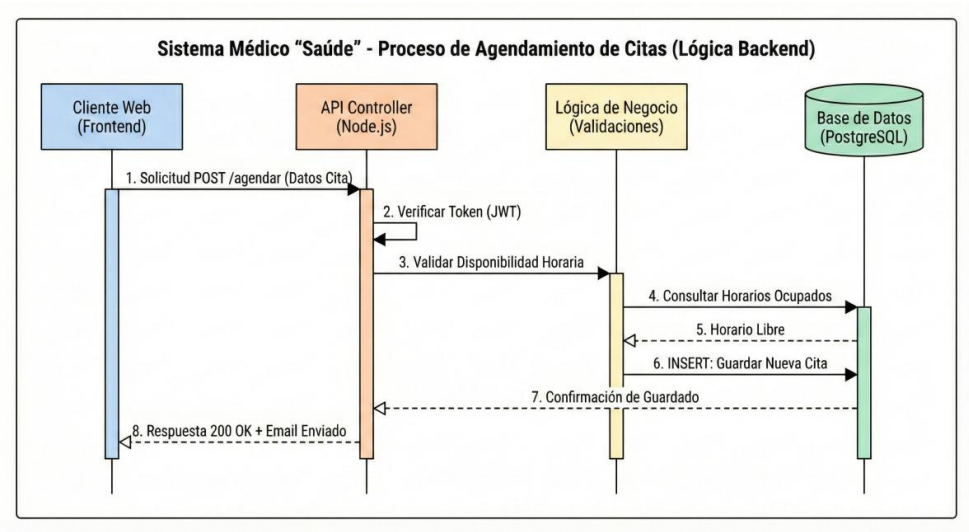


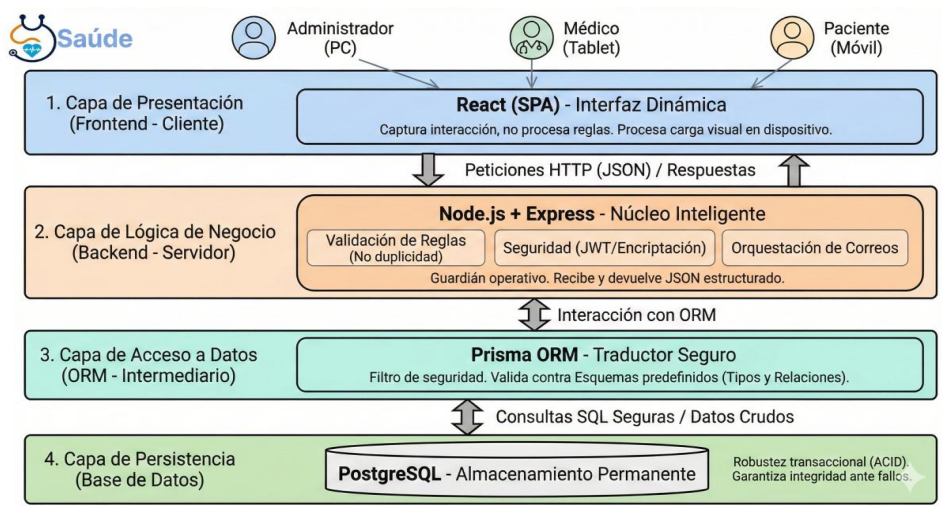
Figura 5

Diagrama de lógica interna del servidor.



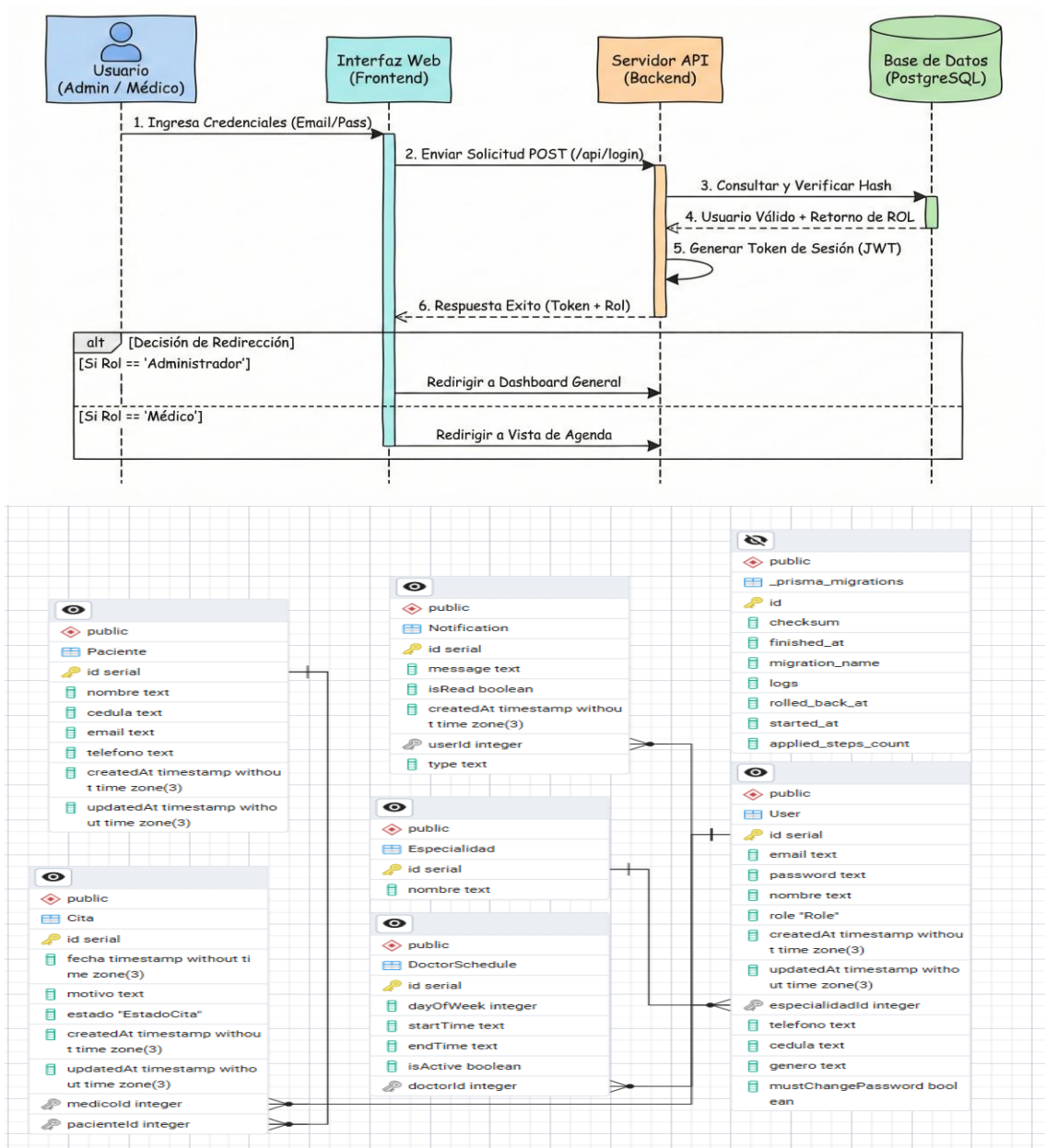
Nota: Diagrama que detalla la lógica interna en el servidor del sistema.

Figura 6.



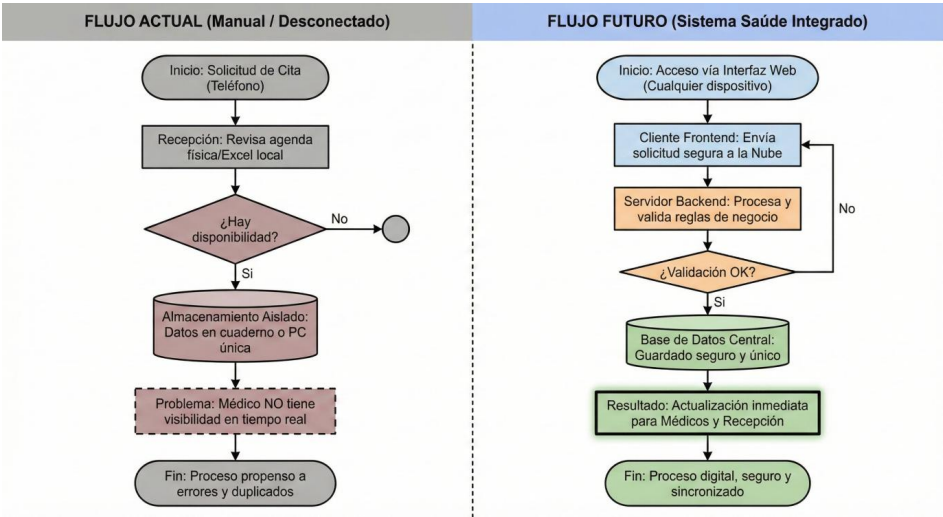
Nota: Estructura arquitectónica en capas del sistema que ilustra la segregación de responsabilidades entre la interfaz de usuario, la lógica de negocio y la persistencia de datos.

Figura 7



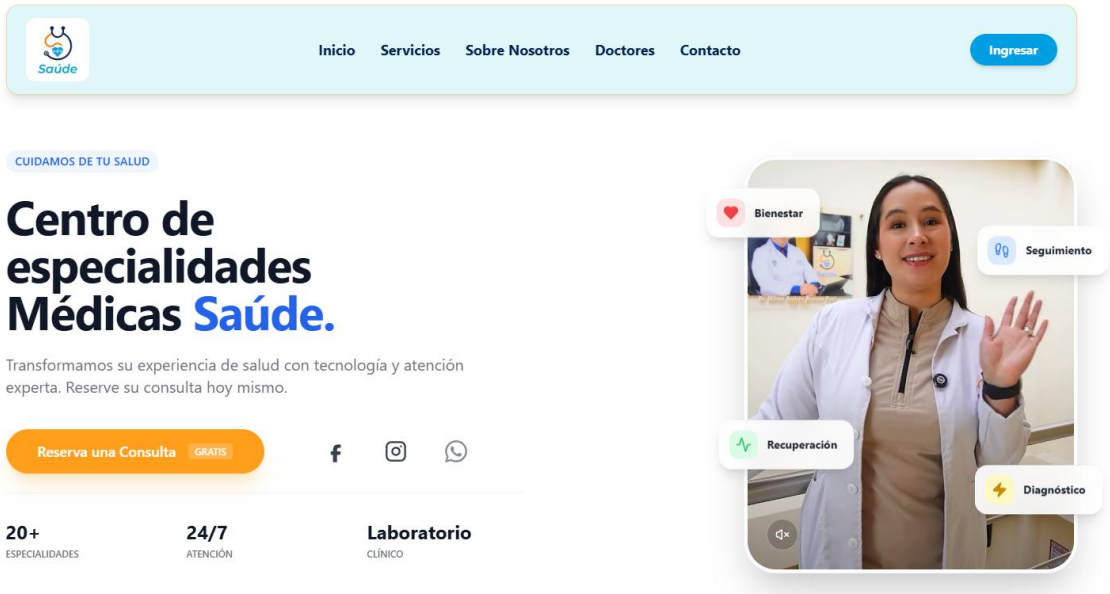
Nota: Demuestra cómo el sistema maneja la seguridad de forma centralizada, pero deriva a los usuarios a sus áreas correspondientes de forma inteligente y el diagrama de la base de datos

Figura 8



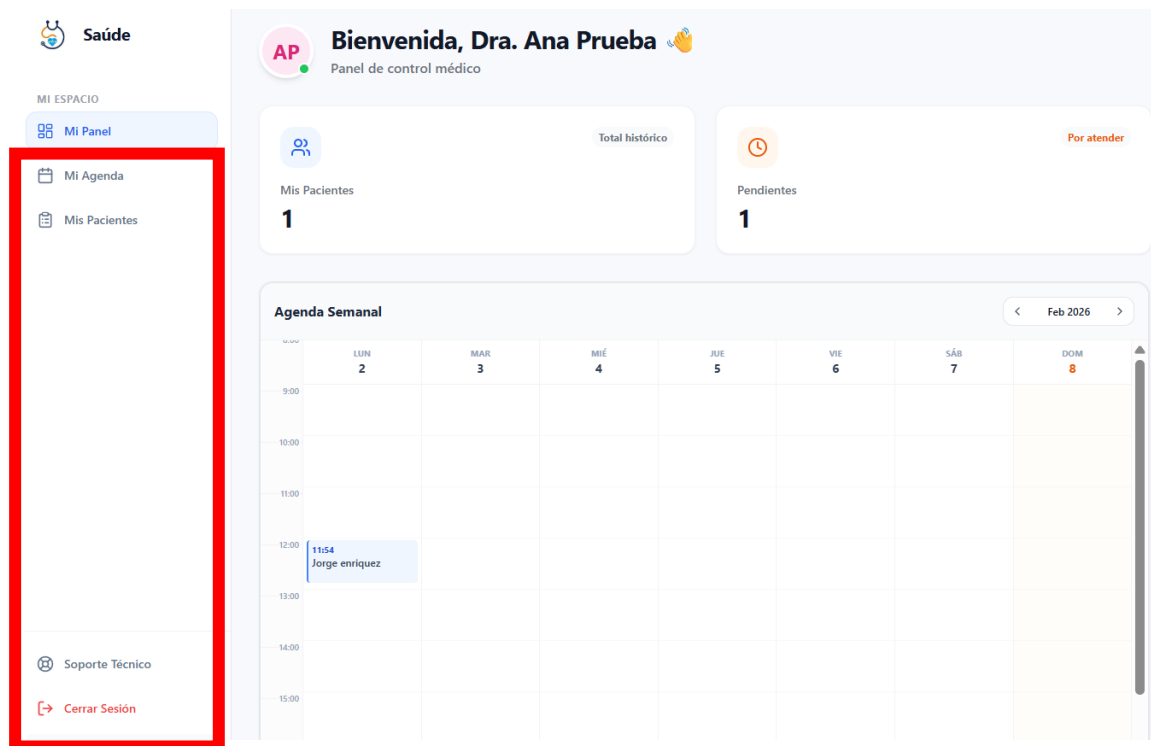
Nota: Diagrama de flujo comparativo. A la izquierda, el proceso manual actual con almacenamiento de datos aislado. A la derecha, el flujo futuro integrado mediante la arquitectura Cliente-Servidor y base de datos

Figura 9



Nota: Interfaz principal

Figura 10



Nota: Interfaz Médico

Figura 11

The screenshot shows the 'Nuevo Paciente' form. It has a title bar with a close button. The form contains three input fields: 'NOMBRE' with the value 'Jorge Yopez', 'CÉDULA' (empty), and 'EMAIL' with the value 'jorge@gmail.com'. A red oval highlights the 'EMAIL' field, which has a validation message 'Completa este campo' (Complete this field). At the bottom, there are two buttons: 'Cancelar' and 'Registrar'.

Nota: Muestra un error preventivo en caso de no añadir un campo, todos son obligatorios y con formato

Figura 12

The screenshot displays a web application for patient management. At the top, there are two modal forms: 'Nuevo Paciente' and 'Editar Paciente'. Both forms have fields for 'NOMBRE', 'CÉDULA', 'EMAIL', and 'TELÉFONO'. The 'Nuevo Paciente' form has a 'Registrar' button, while the 'Editar Paciente' form has a 'Guardar' button. Below these forms is a 'Pacientes' section with a search bar and a '+ Nuevo Paciente' button. The 'Pacientes' section contains a table with columns: 'NOMBRE DEL PACIENTE', 'CONTACTO', 'CÉDULA', 'DOCTOR ASIGNADO', and 'ACCIONES'. The table lists a patient named 'Jorge enriquez' with ID 'P-1003', contact '9000000000' and 'xs@saude.com', and ID '1234567890'. The assigned doctor is 'Ana Mai Pru Urok'. The 'ACCIONES' column shows 'Editar' and 'Eliminar' buttons.

Nota: Muestra la gestión de pacientes, añadir, editar, agendar, eliminar.

Figura 13

The screenshot shows a modal form titled 'Agendar Nueva Cita'. It has fields for 'PACIENTE' (Jorge enriquez (1234567890)), 'DOCTOR' (Dra. Liseth Cali), 'FECHA' (14/01/2026), 'HORA' (11:00 a. m.), and 'MOTIVO' (Ej. Revisión anual...). There are 'Cancelar' and 'Agendar Cita' buttons at the bottom.

Nota: Interfaz de agendamiento de citas

Figura 14



Nota: Catálogo de Especialidades en página principal.

Figura 15

Registrar Nuevo Doctor

NOMBRE COMPLETO
Juliana Lucía Espinel Cortez

CÉDULA
0179874848

GÉNERO
Femenino

EMAIL
jlespinel@saude.com

TELÉFONO
09000087654

ESPECIALIDAD
Seleccione...

Cancelar Crear

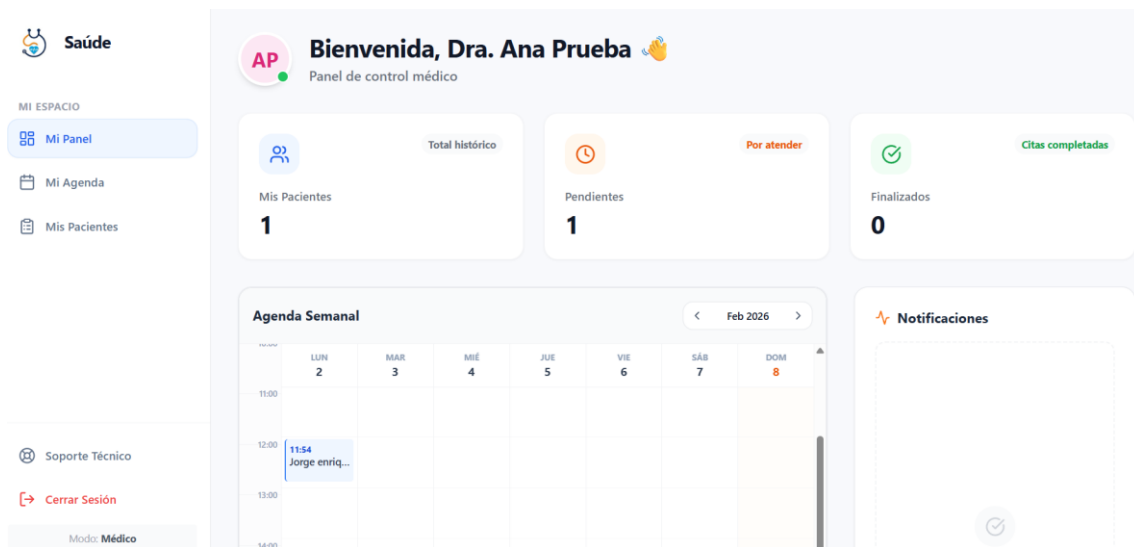
Gestionar Horario
Configura la disponibilidad de Juliana Lucía Espinel Cortez

☒ Lunes	09:00 a. m.	a	05:00 p. m.	✓
☒ Martes	09:00 a. m.	a	05:00 p. m.	✓
☒ Miércoles	09:00 a. m.	a	05:00 p. m.	✓
☒ Jueves	09:00 a. m.	a	05:00 p. m.	✓
☒ Viernes	09:00 a. m.	a	05:00 p. m.	✓
☐ Sábado	09:00 a. m.	a	05:00 p. m.	
☐ Domingo	09:00 a. m.	a	05:00 p. m.	

Cancelar Guardar Horario

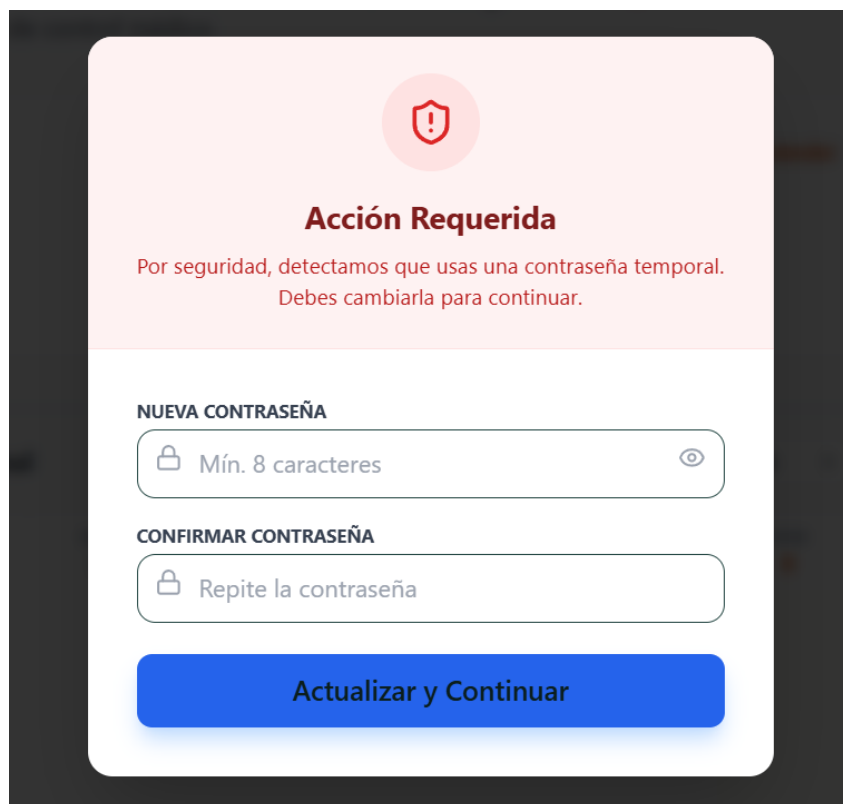
Nota: Interfaz de inclusión de nuevo personal médico

Figura 16



Nota: Muestra la interfaz limpia e intuitiva del rol Médico

Figura 17



Nota: Acción requerida de cambio de contraseña al iniciar sesión por primera

vez