ANEXOS

A.1 MANUAL DEL USUARIO

La presente guía de usuario está meticulosamente diseñada para orientar la operación completa, segura y eficiente del sistema de monitoreo biométrico. Este sistema representa una solución integral que se compone de dos interfaces fundamentales: el dispositivo físico de captura de signos vitales, y el portal web de gestión de datos, el cual actúa como la capa de registro, análisis y seguimiento clínico. Para garantizar la trazabilidad, la integridad y la seguridad de las mediciones recopiladas, el sistema opera bajo un principio fundamental: todo dato biométrico capturado debe estar vinculado de manera verificable a un perfil de usuario registrado. Por consiguiente, el usuario debe seguir un flujo de uso estricto y secuencial, donde la configuración y registro digital en la plataforma web constituyen un prerrequisito obligatorio e ineludible antes de intentar utilizar el dispositivo físico por primera vez.

La estructura de este manual refleja dicho flujo de operación y se divide en dos secciones principales que deben ser abordadas en estricto orden: primero, la guía de operación de la plataforma web, y posteriormente, la guía de operación del dispositivo físico.

PÁGINA WEB

El portal de gestión web es la herramienta central, el núcleo digital y la interfaz principal de visualización y análisis de todo el sistema. Esta plataforma es accesible a través de la dirección https://frontend-pulsioximetro.vercel.app/. Su propósito es múltiple, incluyendo el registro de usuarios, la asignación y gestión de roles y el análisis histórico de las mediciones que han sido guardadas en la base de datos por el dispositivo físico. El portal resulta vital para permitir la visualización de tendencias a largo plazo, la gestión centralizada de perfiles, la administración y supervisión de la estructura del sistema por parte de los administradores, y el seguimiento oportuno de alertas sanitarias por parte del personal médico autorizado. Por esta razón, el primer paso para operar el dispositivo físico es completar la autenticación en el portal web.

PROCESO DE REGISTRO Y AUTENTICACIÓN

La utilización completa de la plataforma de gestión comienza con la necesidad de registrar una nueva cuenta. Para iniciar este proceso, el usuario debe acceder a la página de inicio y seleccionar la opción "Registrarse aquí".



Figura 106, Pantalla de inicio resaltando el botón "Registrarse aquí". [Elaboración propia]

Esta selección redirige al usuario a la página de registro, en la cual es necesario completar todos los datos solicitados para proceder con la creación de la cuenta.

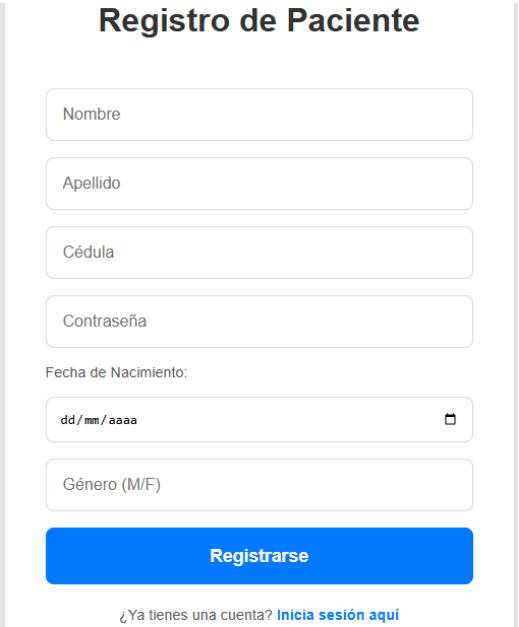


Figura 107, Pantalla de registro de usuario. [Elaboración propia]

Es preciso recalcar dos validaciones críticas del sistema en esta sección: primero, el campo del número de documento sólo admite caracteres numéricos; y segundo, el sistema previene la creación de cuentas con números de cédula que ya hayan sido registrados previamente.

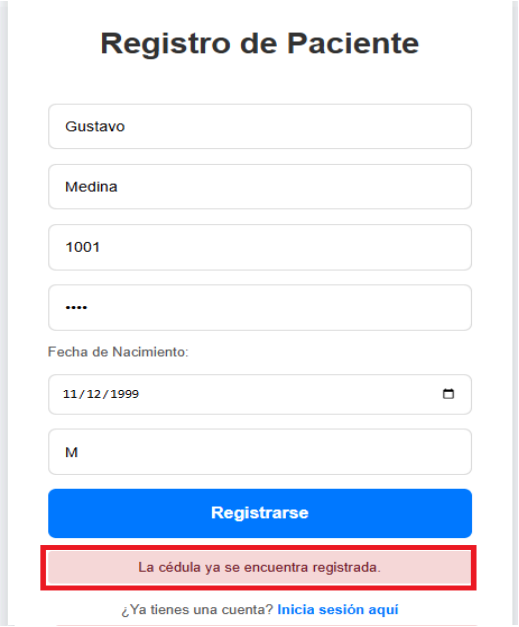


Figura 108, Pantalla de registro mostrando mensaje de error por documento ya registrado. [Elaboración propia]

Una vez que la cuenta ha sido creada con éxito, la plataforma indicará un mensaje de confirmación y automáticamente retornará a la pantalla de inicio de sesión.

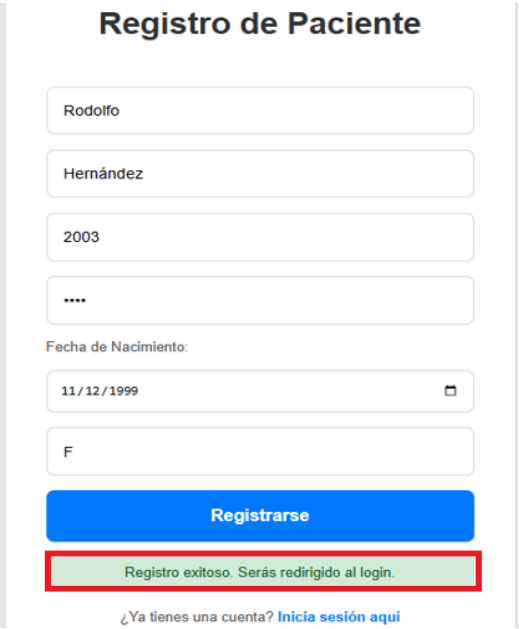


Figura 109, Pantalla de registro con mensaje de éxito. [Elaboración propia]

Para iniciar sesión, el usuario debe ingresar el número de documento con el que se registró la cuenta y la contraseña asociada. Si el sistema detecta credenciales incorrectas, aparecerá un mensaje en pantalla indicando un error en el usuario o la contraseña.



Figura 110, Pantalla de inicio de sesión mostrando mensaje de error por credenciales incorrectas. [Elaboración propia]

Si las credenciales son correctas, el sistema verifica la cuenta y accede a la pantalla principal para visualizar el perfil, cuya estructura y funcionalidad varían en función del rol asignado al usuario.

PAGINA PARA EL PACIENTE

Al acceder con una cuenta que posee el rol de paciente, la plataforma redirige a una pantalla específica diseñada para el monitoreo personal y la gestión de datos, la cual está organizada en cinco paneles principales.



Figura 111, Pantalla general del perfil del paciente. [Elaboración propia]

El primer componente es un banner superior que da un mensaje de bienvenida al paciente, identificándolo por su nombre y número de documento. Este panel incluye un botón que permite cerrar la sesión y retornar a la pantalla de inicio.



Figura 112, Panel superior de bienvenida para el perfil de los pacientes y botón de cierre de sesión. [Elaboración propia]

El segundo panel se encuentra ubicado a la izquierda y tiene la función de mostrar la información básica del perfil del paciente. Adicionalmente, contiene un botón que facilita el acceso a la pantalla de edición del perfil.

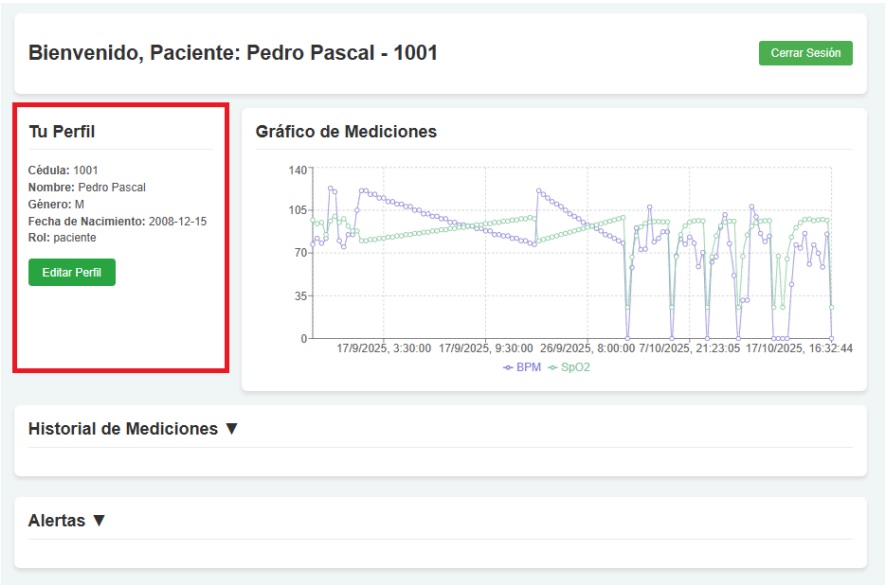


Figura 113, Panel izquierdo del perfil del paciente. [Elaboración propia]

La pantalla de edición permite al usuario modificar sus datos personales, con la excepción del rol y del número de documento. En esta interfaz, se encuentran dos botones de acción: uno para cancelar los cambios y otro para guardarlos y confirmarlos.

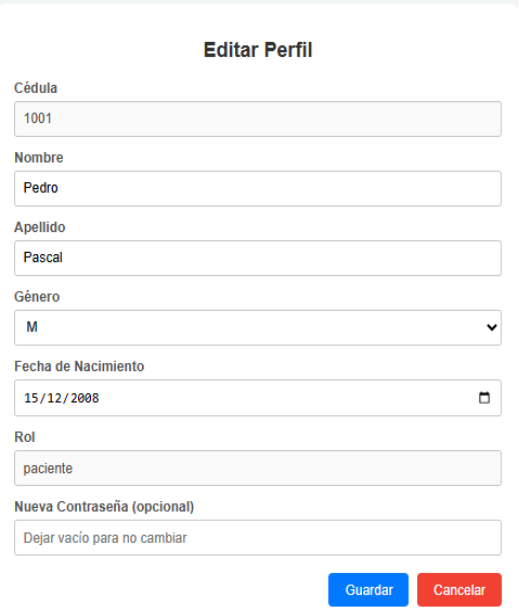


Figura 114, Pantalla de cambio de datos de perfil. [Elaboración propia]

El tercer panel está localizado en el centro y a la derecha de la pantalla. Se caracteriza por ser una gráfica de tendencia de las mediciones registradas, ayudando al paciente a comprender visualmente el comportamiento de sus signos vitales en un periodo de tiempo determinado.



Figura 115, Panel derecho con gráfica de tendencia de datos biométricos. [Elaboración propia]

El cuarto panel, situado en la primera sección inferior, es la tabla de mediciones. Esta tabla almacena el registro histórico de todas las lecturas tomadas con el pulsioxímetro, permitiendo visualizar el ID del registro, los valores de BPM y SpO2, así como la fecha y hora exactas de cada registro.



Figura 116, Panel de tabla de registros médicos. [Elaboración propia]

Para facilitar la búsqueda precisa de registros, la tabla está equipada con varios filtros. Las consultas pueden realizarse por rangos de fecha definiendo una fecha inicial y final o por rangos de valores estableciendo valores mínimos y máximos para BPM y SpO2. Todos estos filtros son combinables para búsquedas más eficientes, y se incluye un botón para eliminar los filtros y restablecer la visualización por orden temporal descendente.

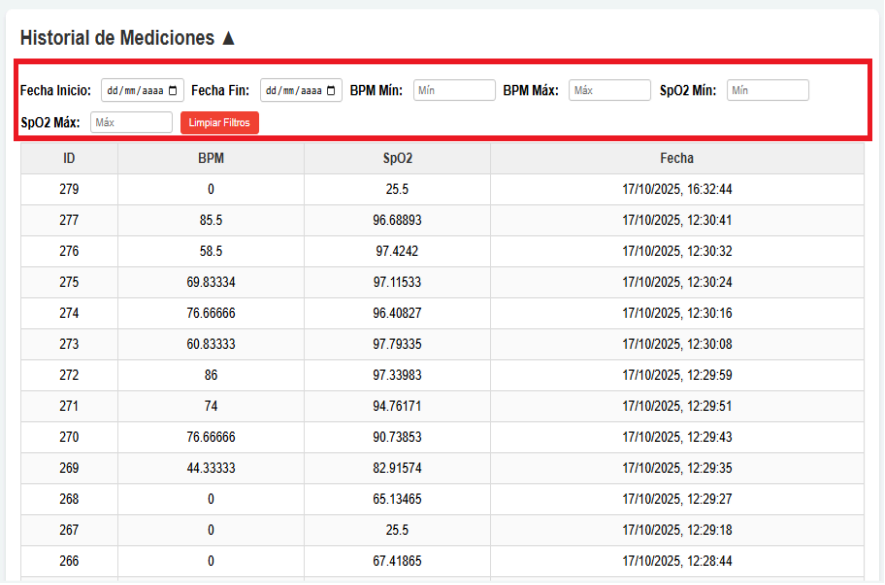


Figura 117, Filtros de la tabla de registros médicos. [Elaboración propia]

Para optimizar la visualización y mejorar la fluidez de la pantalla, la tabla cuenta con un botón adicional el cual le permite contraerse y expandirse, lo que es útil para visualizar el último panel.

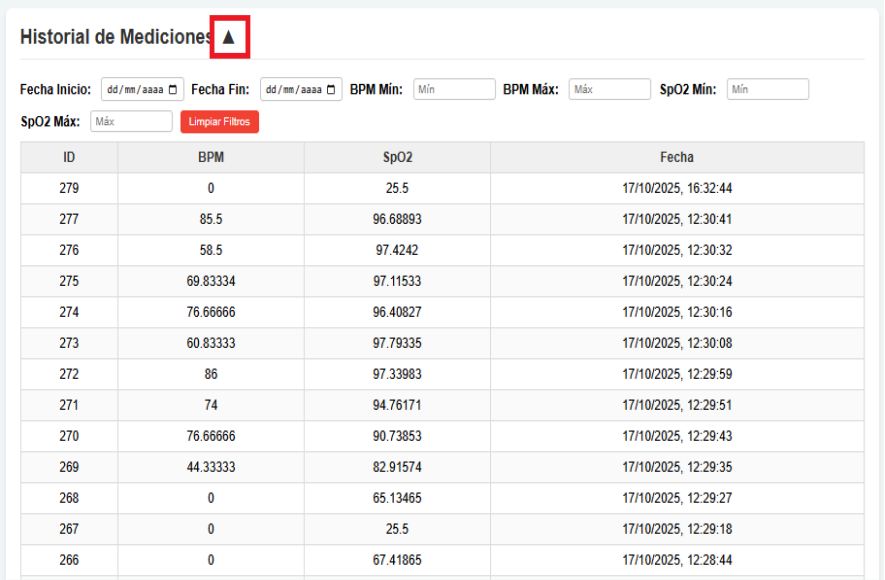


Figura 118, Botón de contracción de la tabla de registros médicos. [Elaboración propia]

Finalmente, el quinto y último panel es la tabla de alertas médicas, ubicada en la parte más inferior de la pantalla. Estas alertas son notificaciones automáticas generadas cuando una medición registra características fuera de los rangos clínicos establecidos, señalando un posible problema de salud. La tabla detalla el ID de la alerta, el ID de la medición de referencia, la fecha, hora, los datos biométricos, un código de color para identificar el tipo de alerta, un mensaje descriptivo y un estado de lectura que indica si el personal médico ha revisado la alerta.



Figura 119, Panel inferior de tabla de alertas médicas. [Elaboración propia]

De forma análoga, esta tabla también incorpora filtros para facilitar la búsqueda, permitiendo realizar consultas por rangos de fechas, por el tipo de alerta y por el estado de lectura para que el paciente sepa qué alertas tienen seguimiento. Estas consultas permiten aplicar múltiples filtros para obtener resultados precisos, y se incluye un botón para eliminar los filtros implementados.

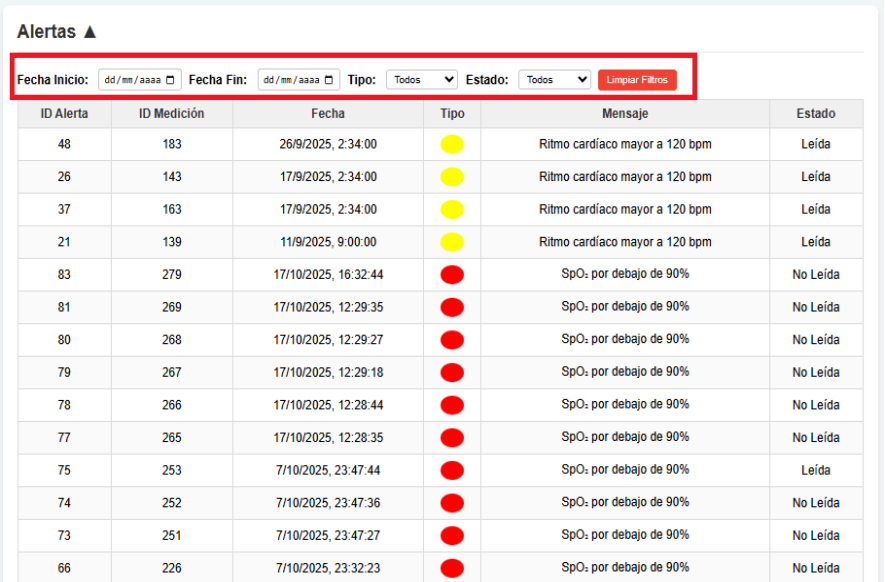


Figura 120, Filtros del panel inferior de la tabla de alertas médicas. [Elaboración propia]

Por último, la tabla de alertas cuenta con un botón que permite contraerse y expandirse con el fin de mejorar la comodidad y la visualización de la pantalla principal del paciente.

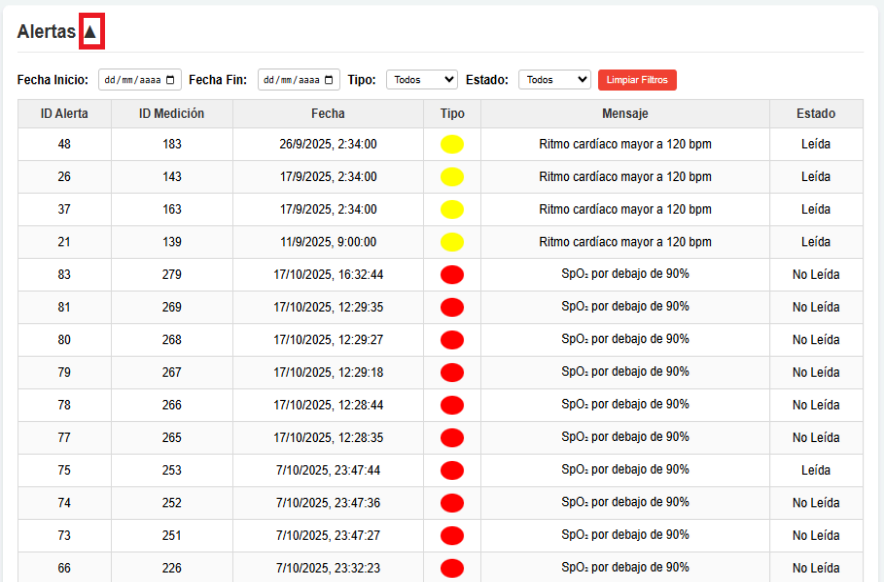


Figura 121, Botón de contracción de la tabla de alertas médicas. [Elaboración propia]

PAGINA PARA EL MÉDICO

Al acceder con una cuenta que posee el rol de médico, la plataforma redirige a una interfaz de gestión diseñada para el monitoreo, seguimiento médico y la gestión de datos de todos los pacientes. Esta pantalla principal se organiza en tres paneles funcionales.

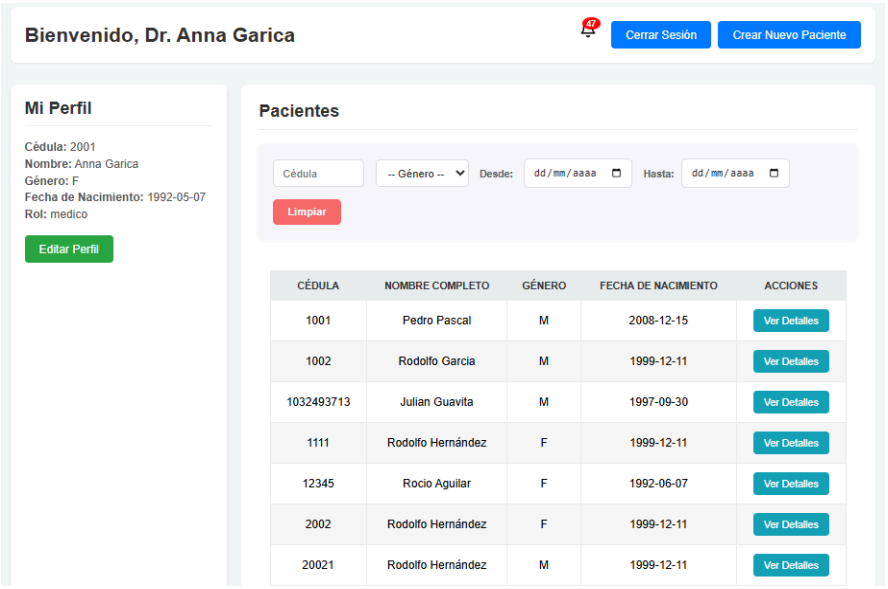


Figura 122, Pantalla general del perfil de médico. [Elaboración propia]

El primer componente es un banner superior que proporciona un mensaje de bienvenida, identificando al médico por su nombre. Este panel es un centro de control rápido, ya que incluye un botón de notificaciones que permite visualizar la cantidad y el detalle de las alertas de los pacientes; otro botón que facilita el acceso a la pantalla para la creación de nuevos pacientes; y un último botón que permite cerrar la sesión y retornar a la pantalla de inicio.

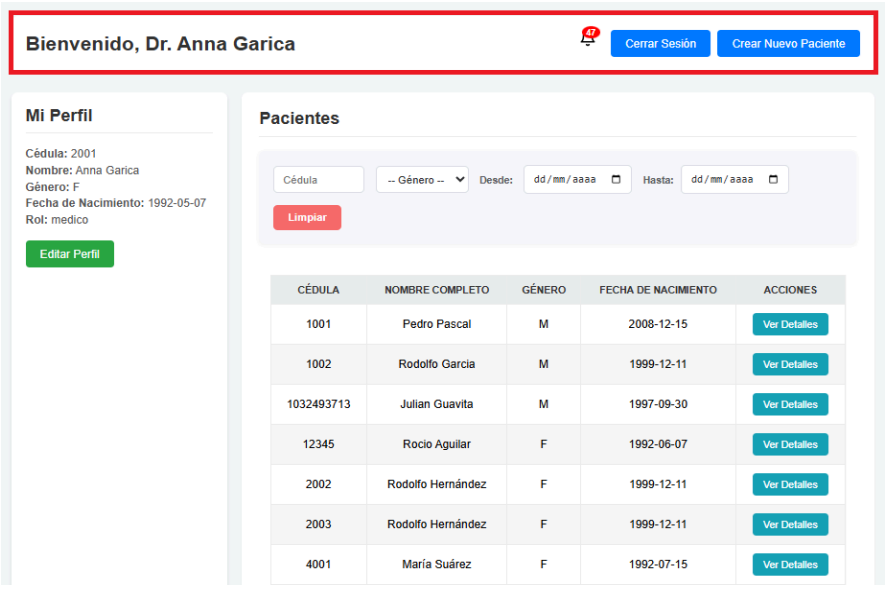


Figura 123, Panel superior para el perfil de los médicos. [Elaboración propia]

La función de registro de nuevos pacientes está diseñada para que los médicos faciliten la incorporación de usuarios, especialmente aquellos que tienen mayor dificultad en el manejo de la tecnología. Al igual que el registro inicial de cuentas, es necesario completar todos los datos solicitados para proceder con la creación del nuevo paciente.

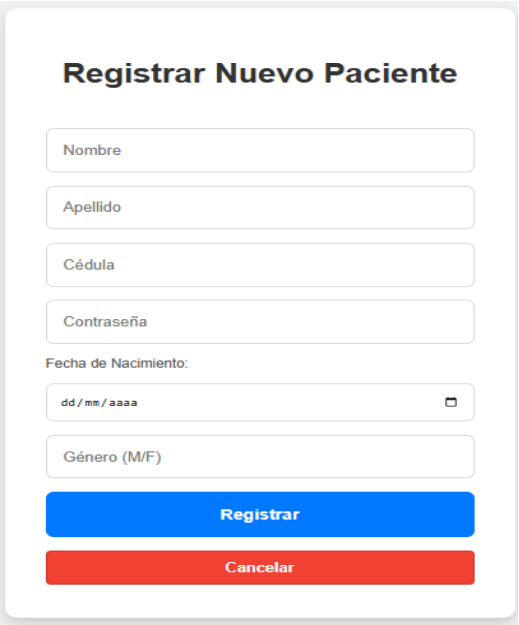


Figura 124, Pantalla de registro de nuevos pacientes. [Elaboración propia]

El sistema mantiene dos validaciones críticas durante este proceso: primero, el campo del número de documento sólo admite caracteres numéricos; y segundo, se previene la creación de cuentas con números de cédula que ya hayan sido previamente registrados.

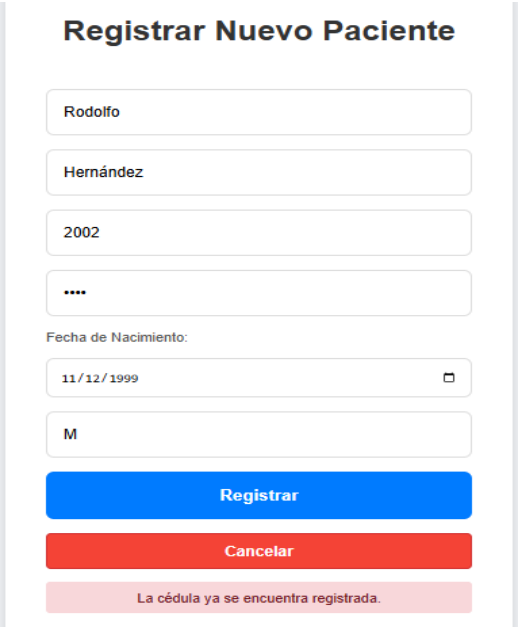


Figura 125, Pantalla de registro mostrando mensaje de error por documento ya registrado. [Elaboración propia]

Una vez que el nuevo paciente ha sido registrado con éxito, la plataforma indicará un mensaje de confirmación y automáticamente retornará a la pantalla principal del médico.

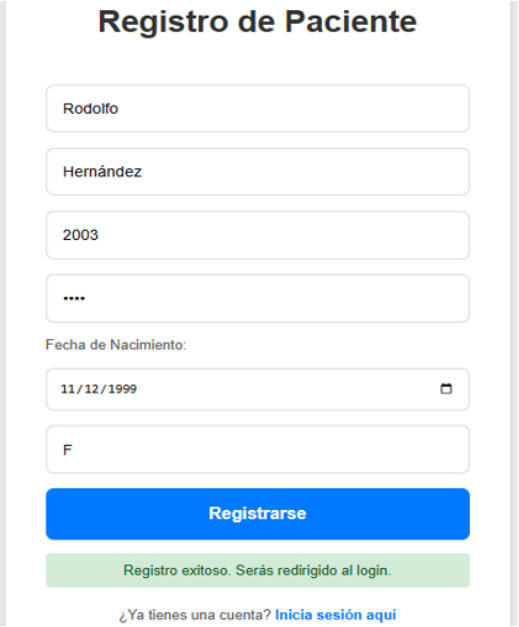


Figura 126, Pantalla de registro de nuevo paciente con mensaje de éxito. [Elaboración propia]

El segundo panel se encuentra ubicado a la izquierda y cumple la función de mostrar la información básica del perfil del médico. Este panel contiene, además, un botón de acceso a la pantalla de edición del perfil.

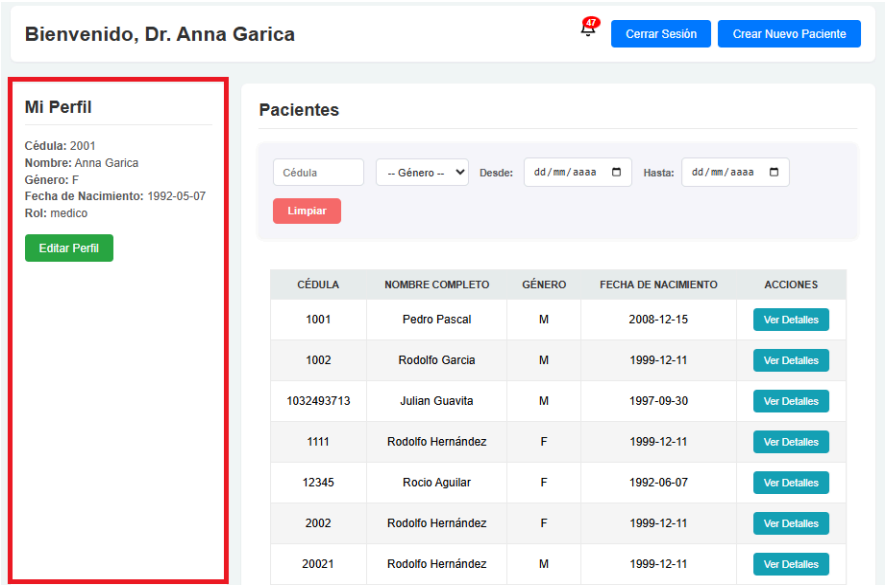


Figura 127, Panel izquierdo del perfil del médico. [Elaboración propia]

La pantalla de edición permite al médico modificar sus datos personales, con la única excepción del rol. En esta interfaz, se encuentran dos botones de acción: uno para cancelar los cambios y otro para guardarlos y confirmarlos.

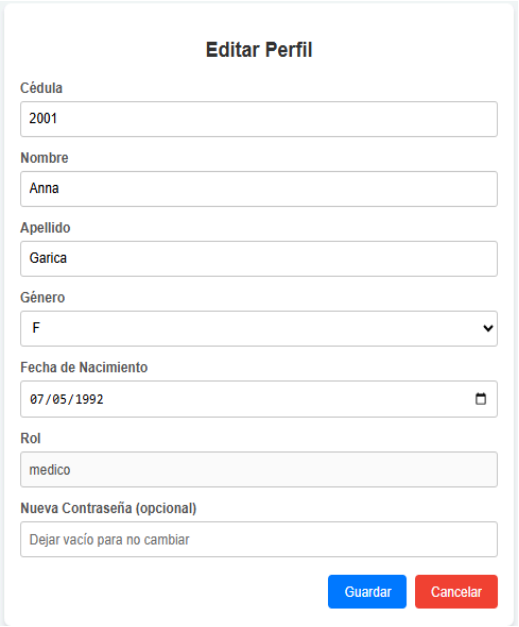


Figura 128, Pantalla de cambio de datos de perfil. [Elaboración propia]

Al igual que en el registro de usuarios, hay dos validaciones críticas del sistema: el campo del número de documento solo admite caracteres numéricos y el sistema previene la duplicidad de documentos ya registrados.

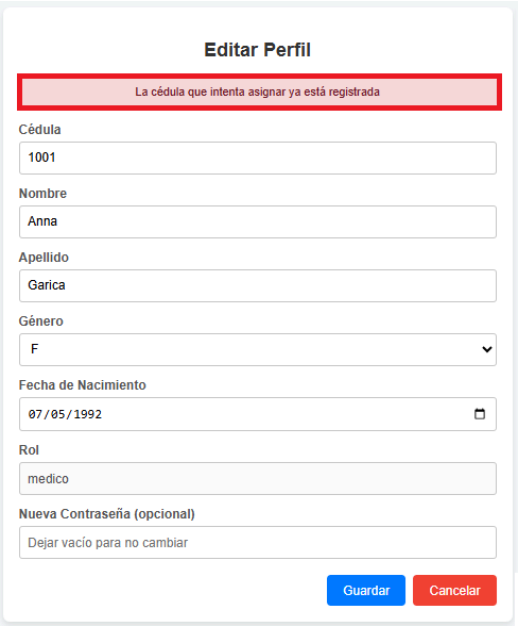


Figura 129, Pantalla de edición de perfil mostrando mensaje de error por documento ya registrado. [Elaboración propia]

Una vez que los datos se editan y guardan con éxito, la plataforma indica un mensaje de confirmación y retorna a la pantalla principal del médico.

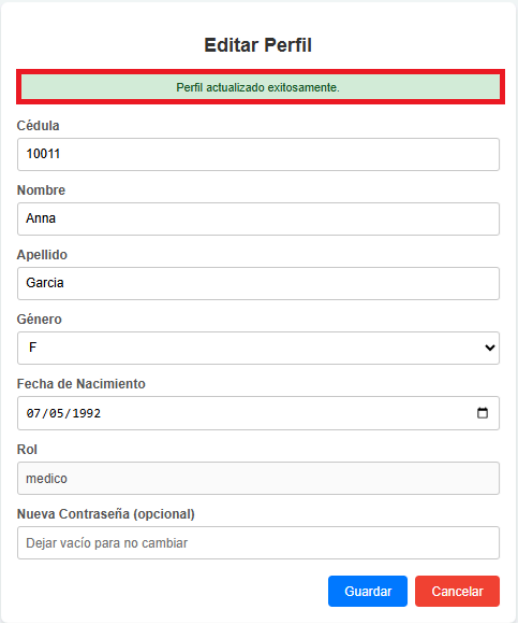


Figura 130, Pantalla de edición del perfil con mensaje de éxito. [Elaboración propia]

En el caso específico de que el cambio incluya la modificación del número de documento o la contraseña del médico y el cambio sea exitoso, una ventana emergente informará al médico que la información de seguridad ha sido cambiada y que debe volver a iniciar sesión para continuar.

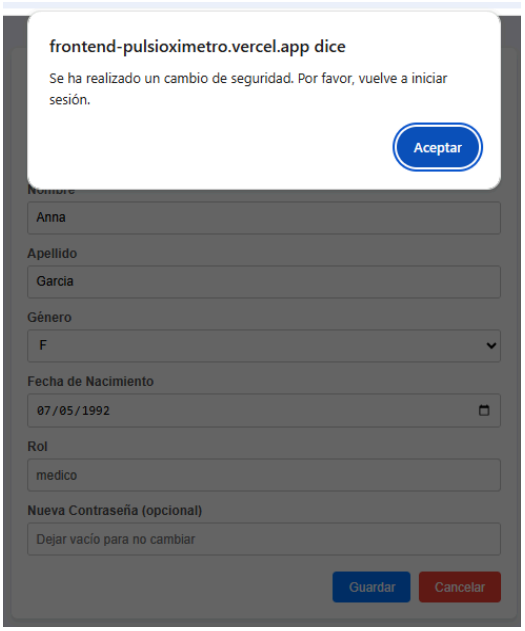


Figura 131, Pantalla de edición del perfil con mensaje de éxito y regreso al inicio de sesión. [Elaboración propia]

El tercer y más importante panel se encuentra localizado en el centro y a la derecha de la pantalla. Se caracteriza por ser la tabla de pacientes registrados, funcionando como una lista exhaustiva que muestra la cédula, el nombre, el género y la fecha de nacimiento de todos los usuarios, permitiendo al médico dar seguimiento a cada uno de ellos.

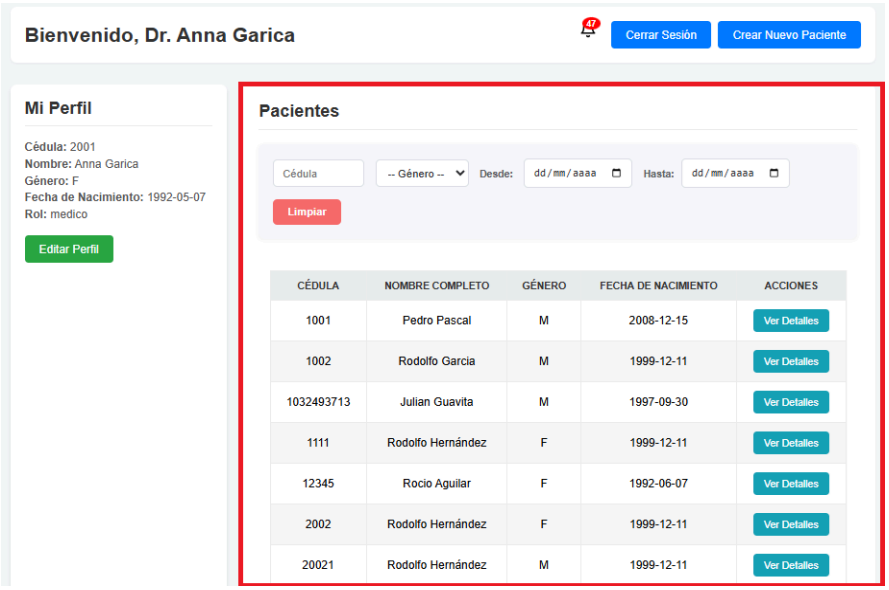


Figura 132, Panel derecho tabla de pacientes. [Elaboración propia]

Para facilitar la búsqueda precisa de pacientes, la tabla está equipada con varios filtros. Las consultas pueden realizarse por rangos de fecha de nacimiento definiendo una fecha mínima y máxima o por número de documento. Estos filtros son combinables, y se incluye un botón para eliminar los filtros y restablecer la visualización por orden de cédula ascendente.

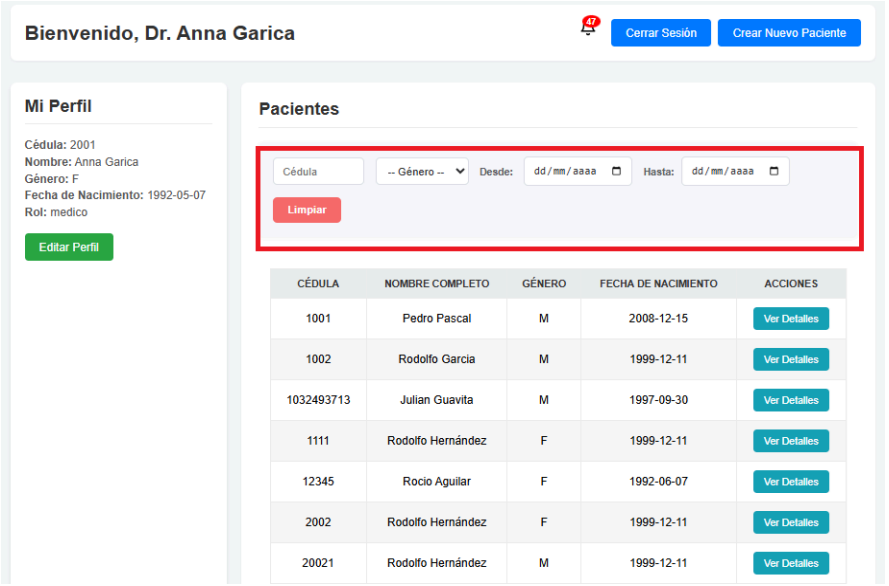


Figura 133, Filtros de la tabla de pacientes. [Elaboración propia]

Finalmente, el registro de cada paciente en la tabla incluye un botón de acción para ver los detalles completos de ese usuario, lo que dirige a la plataforma a una vista de historial clínico específica.

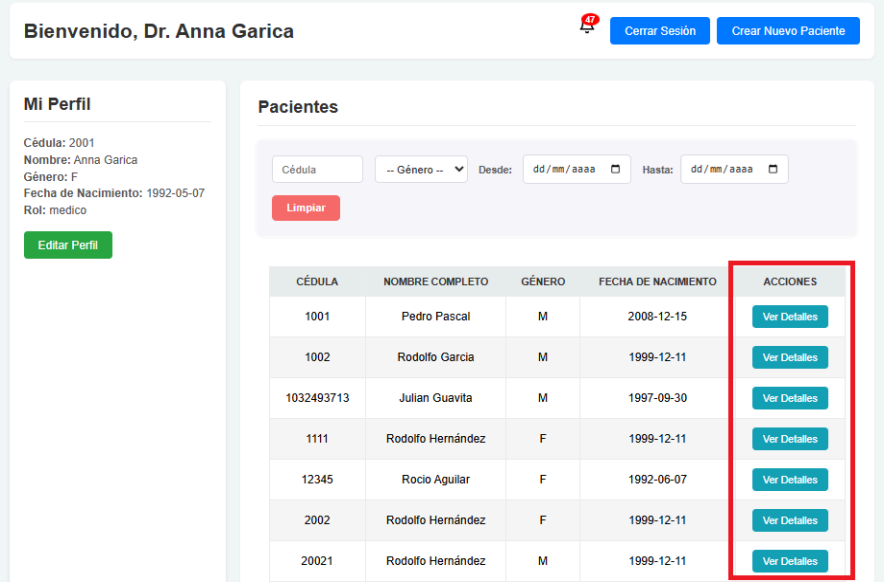


Figura 134, Botón para acceder a la información detallada de los pacientes. [Elaboración propia]

Al usar el botón de ver detalles, la plataforma redirige a otra pantalla, organizada en cinco paneles, donde se puede apreciar tanto el perfil básico como el historial clínico completo del paciente.

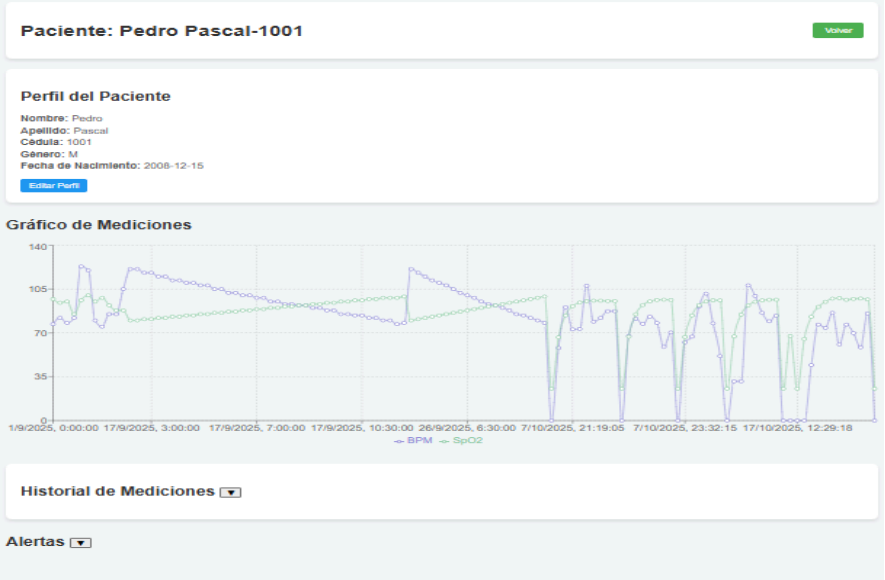


Figura 135, pantalla principal al acceder al botón de la vista de detalles de los pacientes. [Elaboración propia]

El primer componente es un banner superior que da información sobre el paciente, identificándolo por su nombre y número de documento. Este panel incluye un botón que permite volver y retornar a la pantalla principal del médico.



Figura 136, Panel superior del perfil de los pacientes y botón de regreso. [Elaboración propia]

El segundo panel se encuentra justo después del banner y tiene la función de mostrar la información básica del perfil del paciente. Contiene un botón que facilita el acceso a la pantalla de edición del perfil del paciente.

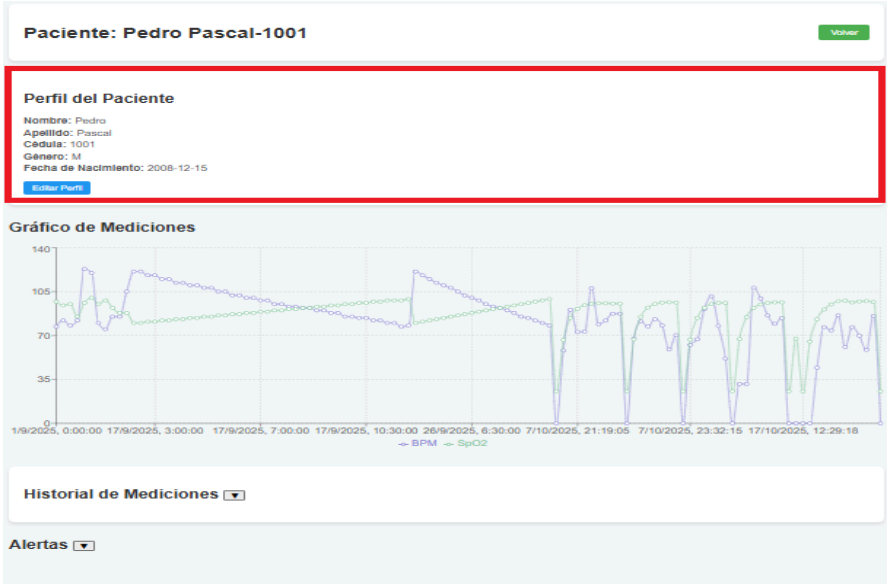


Figura 137, Segundo panel del perfil del paciente. [Elaboración propia]

La pantalla de edición permite al médico modificar los datos personales del paciente, con la excepción del rol. En esta interfaz, se encuentran dos botones de acción: uno para cancelar los cambios y otro para guardarlos y confirmarlos.

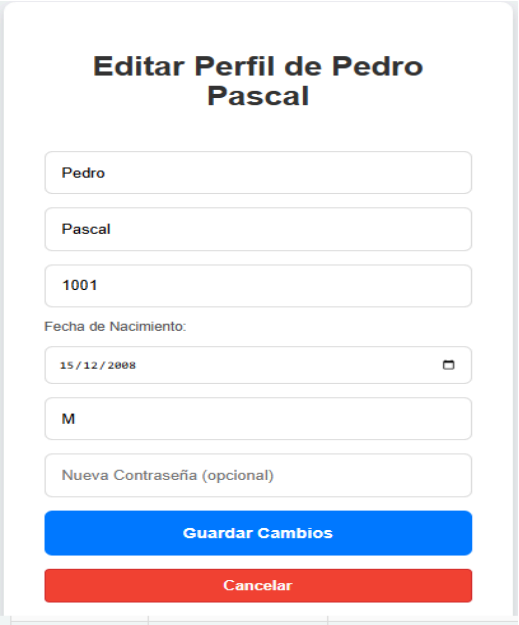


Figura 138, Pantalla de cambio de datos de perfil. [Elaboración propia]

Al igual que en el registro de usuarios, las validaciones críticas del sistema se mantienen: el campo del número de documento solo admite caracteres numéricos, y se previene la duplicidad de documentos ya registrados.



Figura 139, Pantalla de edición de perfil mostrando mensaje de error por documento ya registrado. [Elaboración propia]

Una vez que los datos se editan y guardan con éxito, la plataforma indica un mensaje de confirmación y automáticamente retorna a la pantalla principal del médico.

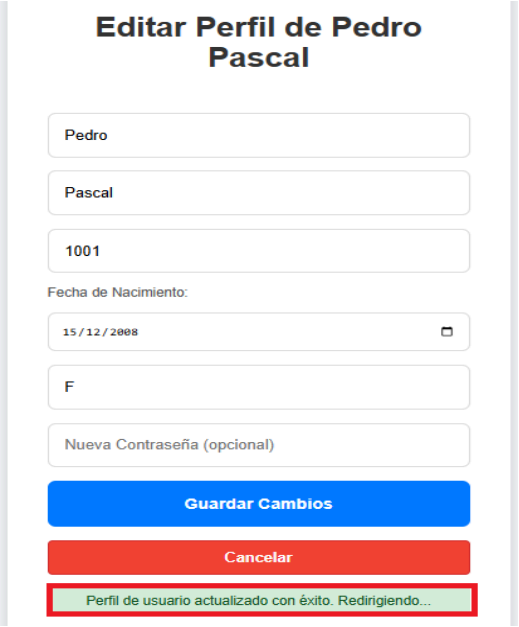


Figura 140, Pantalla de edición del perfil con mensaje de éxito. [Elaboración propia]

En el caso de que el cambio incluya el número de documento del paciente y sea exitoso, una ventana emergente le informará al médico que la información de seguridad ha sido cambiada y que debe volver a la pantalla principal del médico.

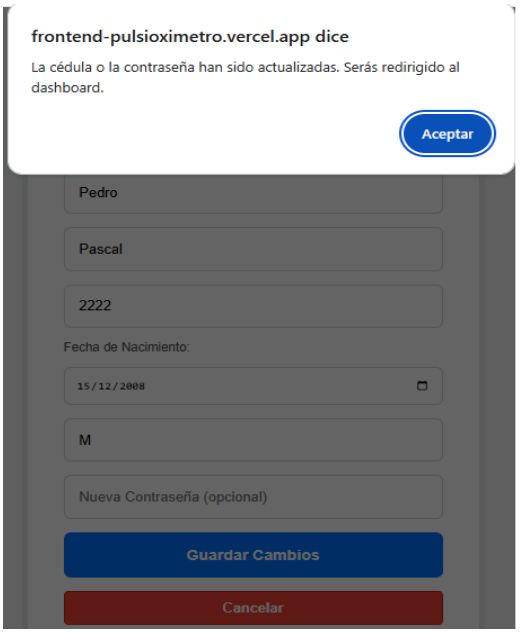


Figura 141, Pantalla de edición del perfil con mensaje de éxito y regreso al inicio de sesión. [Elaboración propia]

El tercer panel está localizado en el centro de la pantalla. Se caracteriza por ser una gráfica de tendencia de las mediciones registradas, lo que ayuda al médico a comprender visualmente el comportamiento de los signos vitales del paciente en un periodo de tiempo determinado.



Figura 142, Tercer panel con gráfica de tendencia de datos biométricos. [Elaboración propia]

El cuarto panel, es la tabla de mediciones. Esta tabla almacena el registro histórico de todas las lecturas tomadas con el pulsioxímetro, permitiendo visualizar el ID del registro, los valores de BPM y SpO2, así como la fecha y hora exactas de cada registro.



Figura 143, Panel de tabla de registros médicos. [Elaboración propia]

Para facilitar la búsqueda precisa de registros, la tabla incorpora varios filtros. Las consultas pueden realizarse por rangos de fecha definiendo una fecha inicial y final o por rangos de valores estableciendo valores mínimos y máximos para BPM y SpO2. Todos estos filtros son combinables y se incluye un botón para eliminar los filtros.

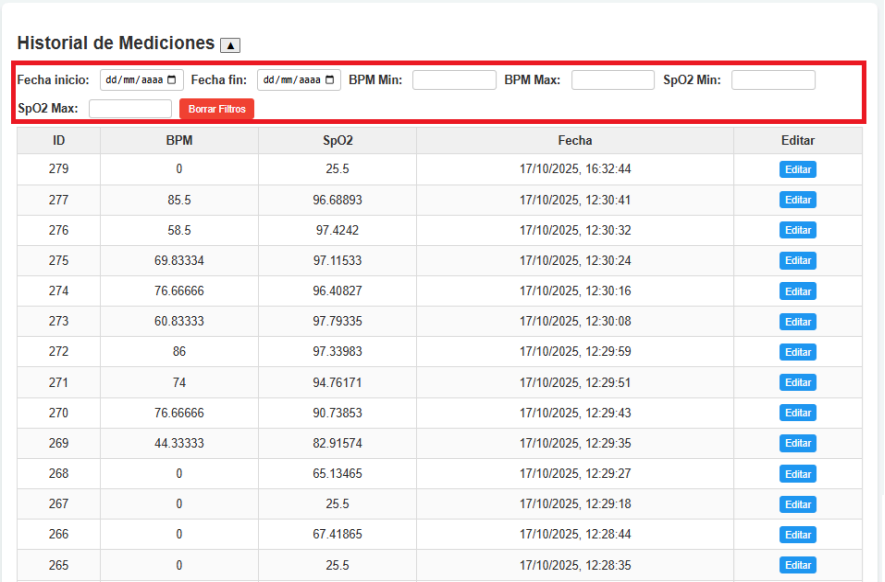


Figura 144, Filtros de la tabla de registros médicos. [Elaboración propia]

Adicionalmente, la tabla incorpora botones de acción para cada registro que permiten la edición de los valores de SpO2 y BPM para corregir pequeños errores que se deban a interferencias o malas lecturas por movimiento.

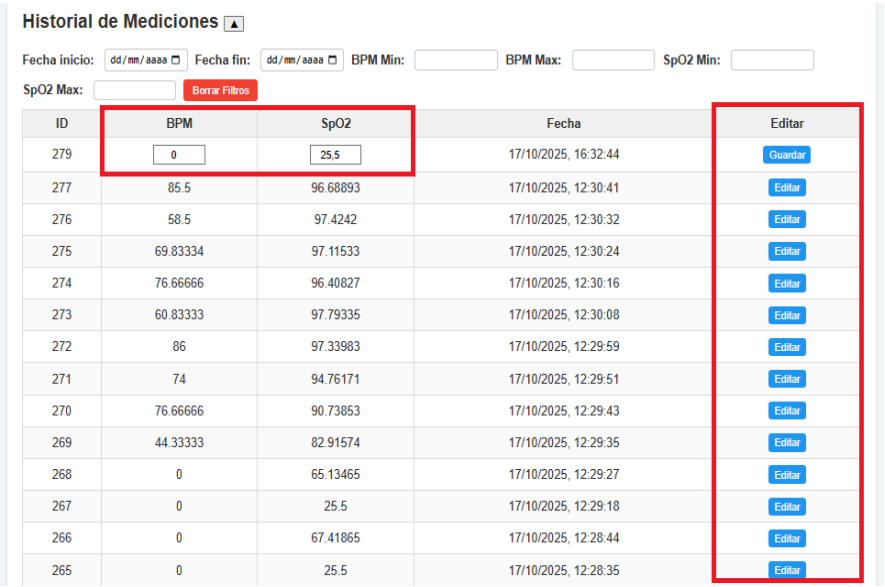


Figura 145, Botón para la edición del spo2 y bpm. [Elaboración propia]

Para optimizar la visualización y mejorar la fluidez de la pantalla, la tabla cuenta con un botón adicional que le permite contraerse y expandirse, lo que es útil para visualizar el último panel.

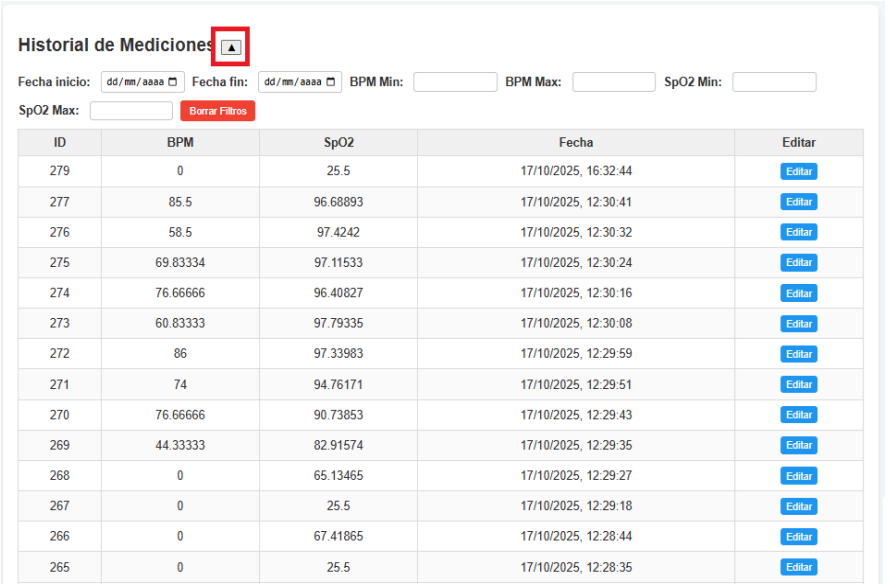


Figura 146, Botón de contracción de la tabla de registros médicos. [Elaboración propia]

Finalmente, el quinto y último panel es la tabla de alertas médicas, ubicada en la parte más inferior de la pantalla. Esta tabla detalla el ID de la alerta, la medición de referencia, la fecha, hora, los datos biométricos, un código de color que identifica el tipo de alerta, un mensaje descriptivo y un estado de lectura que indica el seguimiento por parte del médico.

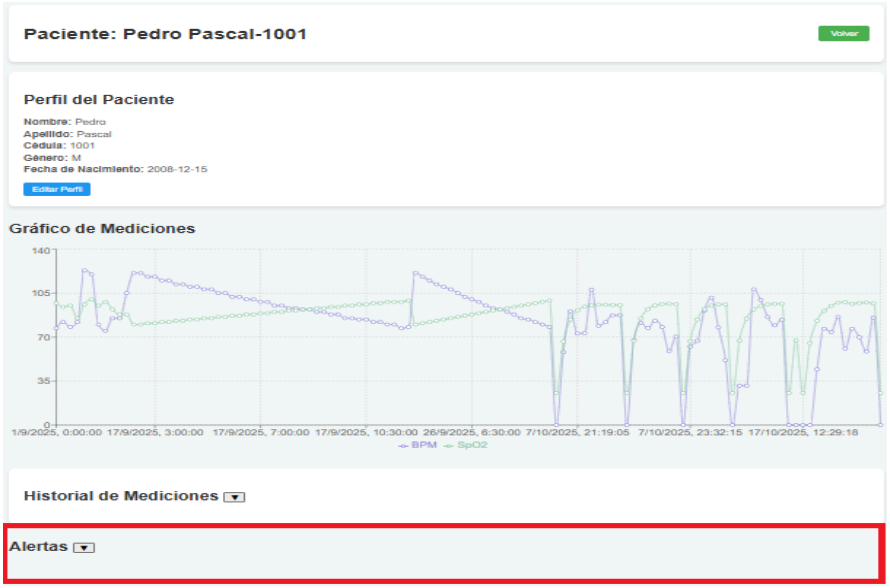
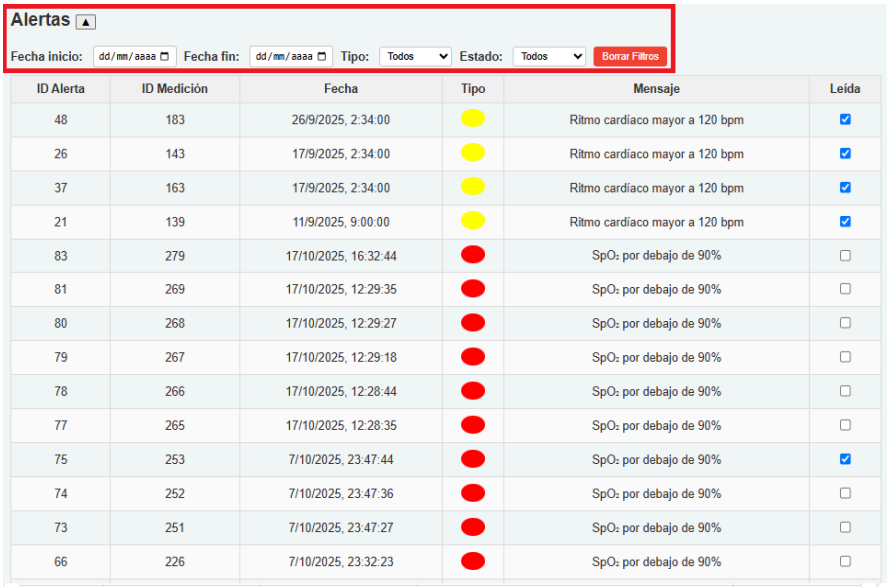


Figura 147, Panel inferior de tabla de alertas médicas. [Elaboración propia]

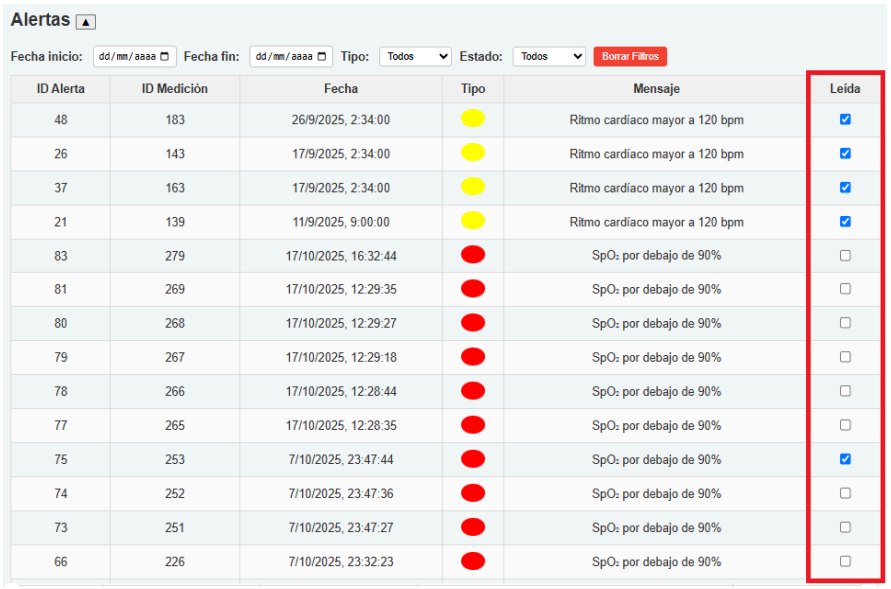
De forma análoga, esta tabla incorpora filtros para facilitar la búsqueda, permitiendo realizar consultas por rangos de fechas, por el tipo de alerta y por el estado de lectura. Las consultas permiten aplicar múltiples filtros para obtener resultados precisos, y se incluye un botón para eliminar los filtros.



ID Alerta	ID Medición	Fecha	Tipo	Mensaje	Leída
48	183	26/9/2025, 2:34:00		Ritmo cardíaco mayor a 120 bpm	☑
26	143	17/9/2025, 2:34:00		Ritmo cardíaco mayor a 120 bpm	☑
37	163	17/9/2025, 2:34:00		Ritmo cardíaco mayor a 120 bpm	☑
21	139	11/9/2025, 9:00:00		Ritmo cardíaco mayor a 120 bpm	☑
83	279	17/10/2025, 16:32:44		SpO_2 por debajo de 90%	☐
81	269	17/10/2025, 12:29:35		SpO_2 por debajo de 90%	☐
80	268	17/10/2025, 12:29:27		SpO_2 por debajo de 90%	☐
79	267	17/10/2025, 12:29:18		SpO_2 por debajo de 90%	☐
78	266	17/10/2025, 12:28:44		SpO_2 por debajo de 90%	☐
77	265	17/10/2025, 12:28:35		SpO_2 por debajo de 90%	☐
75	253	7/10/2025, 23:47:44		SpO_2 por debajo de 90%	☑
74	252	7/10/2025, 23:47:36		SpO_2 por debajo de 90%	☐
73	251	7/10/2025, 23:47:27		SpO_2 por debajo de 90%	☐
66	226	7/10/2025, 23:32:23		SpO_2 por debajo de 90%	☐

Figura 148, Filtros del panel inferior de la tabla de alertas médicas. [Elaboración propia]

Adicionalmente, la tabla incorpora botones de acción para cada alerta que permiten la edición del estado para indicar que el médico ya le dio seguimiento a esas alertas clínicas.



ID Alerta	ID Medición	Fecha	Tipo	Mensaje	Leída
48	183	26/9/2025, 2:34:00		Ritmo cardíaco mayor a 120 bpm	☑
26	143	17/9/2025, 2:34:00		Ritmo cardíaco mayor a 120 bpm	☑
37	163	17/9/2025, 2:34:00		Ritmo cardíaco mayor a 120 bpm	☑
21	139	11/9/2025, 9:00:00		Ritmo cardíaco mayor a 120 bpm	☑
83	279	17/10/2025, 16:32:44		SpO_2 por debajo de 90%	☐
81	269	17/10/2025, 12:29:35		SpO_2 por debajo de 90%	☐
80	268	17/10/2025, 12:29:27		SpO_2 por debajo de 90%	☐
79	267	17/10/2025, 12:29:18		SpO_2 por debajo de 90%	☐
78	266	17/10/2025, 12:28:44		SpO_2 por debajo de 90%	☐
77	265	17/10/2025, 12:28:35		SpO_2 por debajo de 90%	☐
75	253	7/10/2025, 23:47:44		SpO_2 por debajo de 90%	☑
74	252	7/10/2025, 23:47:36		SpO_2 por debajo de 90%	☐
73	251	7/10/2025, 23:47:27		SpO_2 por debajo de 90%	☐
66	226	7/10/2025, 23:32:23		SpO_2 por debajo de 90%	☐

Figura 149, Casilla para la edición del estado de alertas. [Elaboración propia]

Por último, la tabla de alertas cuenta con un botón que permite contraerse y expandirse con el fin de mejorar la comodidad y la visualización de la pantalla del historial médico del paciente.

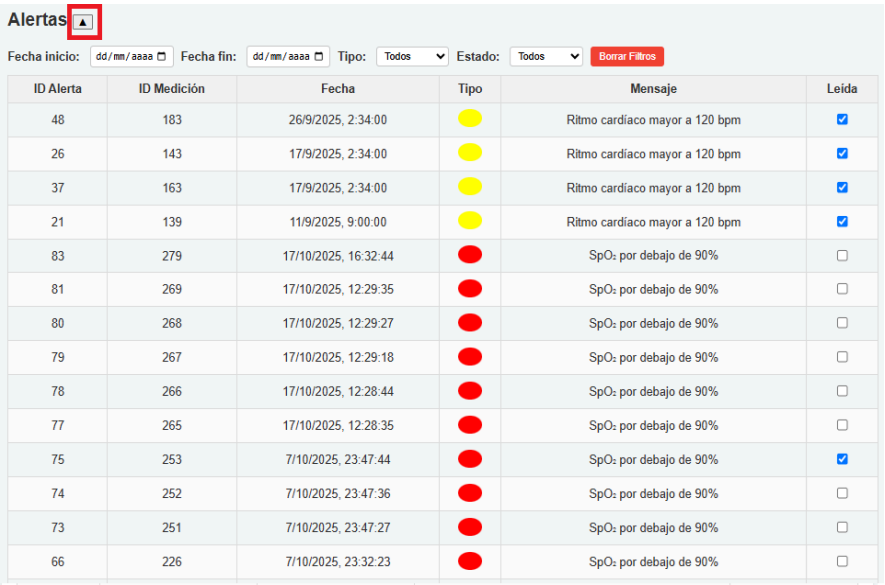


Figura 150, Botón de contracción de la tabla de alertas médicas. [Elaboración propia]

PAGINA PARA EL ADMINISTRADOR

Al acceder con una cuenta que posee el rol de administrador, la plataforma redirige a una interfaz de gestión diseñada para la supervisión general del sistema, el mantenimiento de la integridad de los datos y la gestión de todos los usuarios. Esta pantalla principal se organiza en tres paneles funcionales.

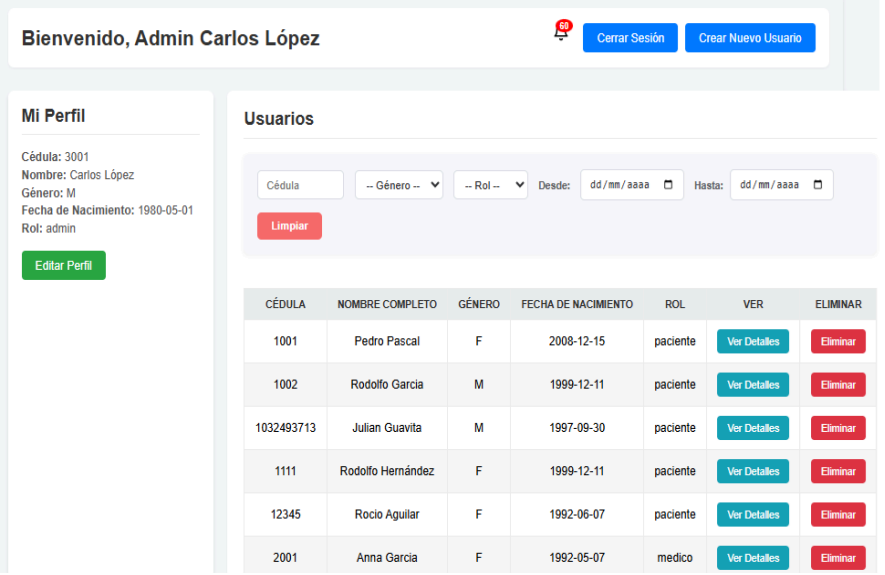


Figura 151, Pantalla general del perfil de administrador. [Elaboración propia]

El primer componente es un banner superior que proporciona un mensaje de bienvenida, identificando al administrador por su nombre. Este panel funciona como un centro de control rápido, ya que incluye un botón de notificaciones que permite visualizar la cantidad y el detalle de las alertas generadas por todos los usuarios; otro botón que facilita el acceso a la pantalla para la creación de nuevos usuarios; y un último botón que permite cerrar la sesión y retornar a la pantalla de inicio.

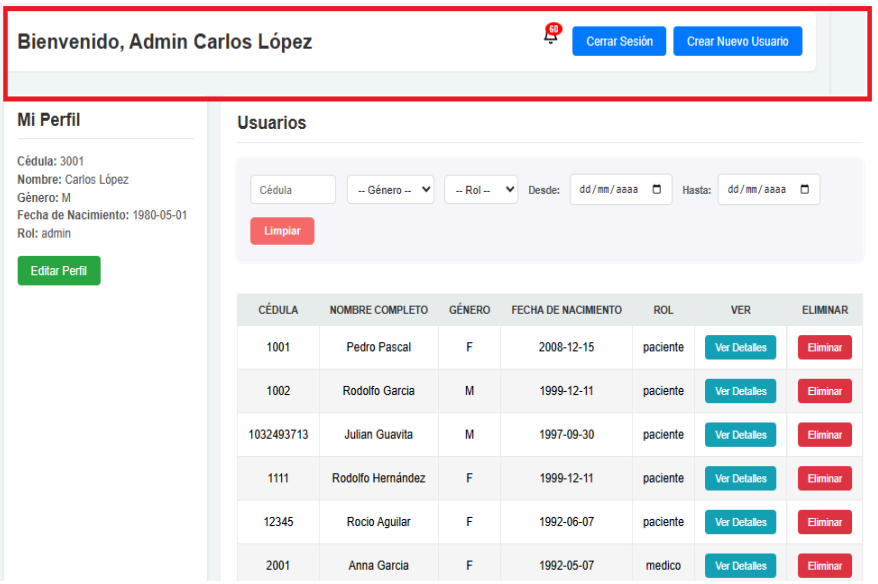


Figura 152, Panel superior para el perfil de los administradores. [Elaboración propia]

La función de registro de nuevos usuarios está diseñada para que los administradores puedan crear cuentas para cualquier rol dentro del sistema: paciente, médico o administrador. Al igual que el registro inicial, es necesario completar todos los datos solicitados para proceder con la creación de la nueva cuenta.

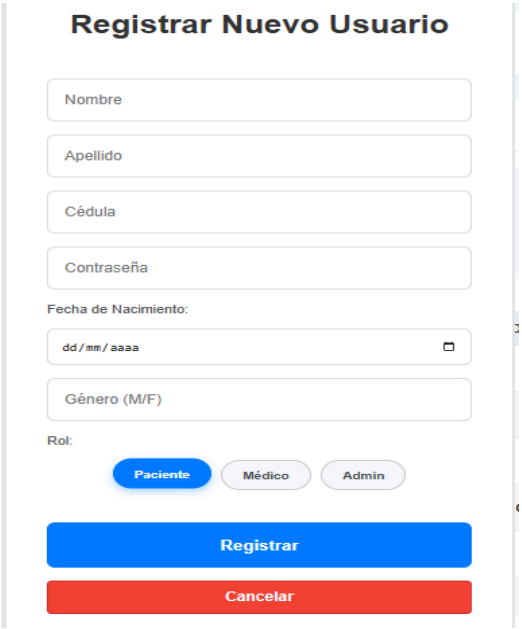


Figura 153, Pantalla de registro de nuevos usuarios. [Elaboración propia]

El sistema mantiene dos validaciones críticas durante este proceso: primero, el campo del número de documento solo admite caracteres numéricos; y segundo, se previene la creación de cuentas con números de cédula que ya hayan sido previamente registrados.

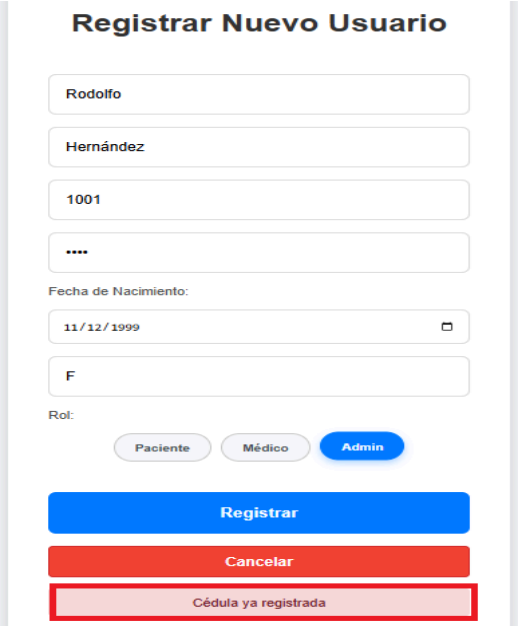


Figura 154, Pantalla de registro mostrando mensaje de error por documento ya registrado. [Elaboración propia]

Una vez que el nuevo usuario ha sido registrado con éxito, la plataforma indicará un mensaje de confirmación y automáticamente retornará a la pantalla principal del administrador.

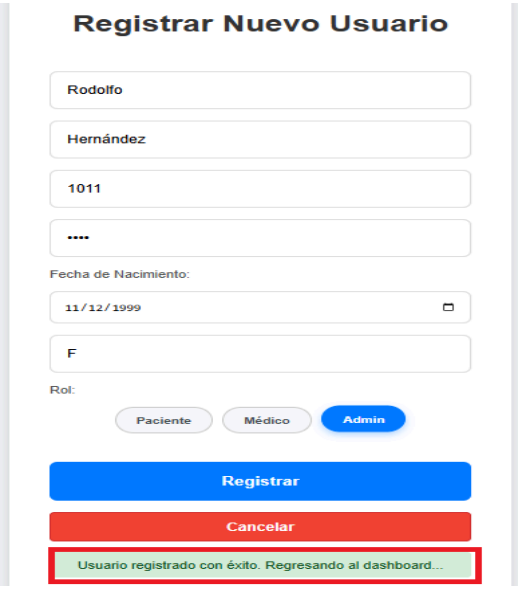


Figura 155, Pantalla de registro de nuevo usuarios con mensaje de éxito. [Elaboración propia]

El segundo panel se encuentra ubicado a la izquierda y cumple la función de mostrar la información básica del perfil del administrador. Contiene, además, un botón de acceso a la pantalla de edición del perfil.

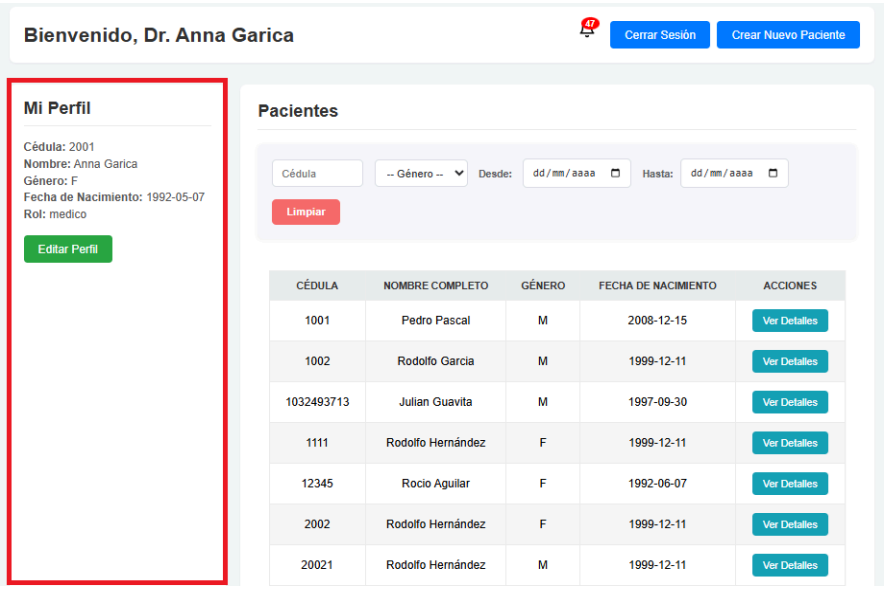


Figura 156, Panel izquierdo del perfil del administrador. [Elaboración propia]

La pantalla de edición permite al administrador modificar sus datos personales sin restricción alguna. En esta interfaz, se encuentran dos botones de acción: uno para cancelar los cambios y otro para guardarlos y confirmarlos.

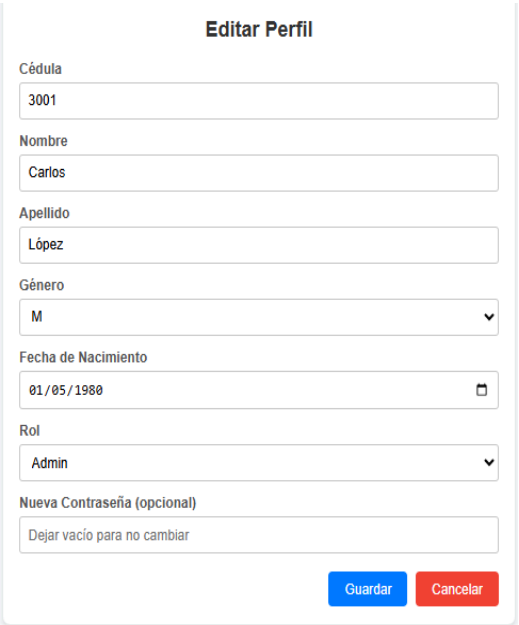


Figura 157, Pantalla de cambio de datos de perfil. [Elaboración propia]

Al igual que en el registro de usuarios, las validaciones críticas del sistema se mantienen: el campo del número de documento solo admite caracteres numéricos y se previene la duplicidad de documentos ya registrados.

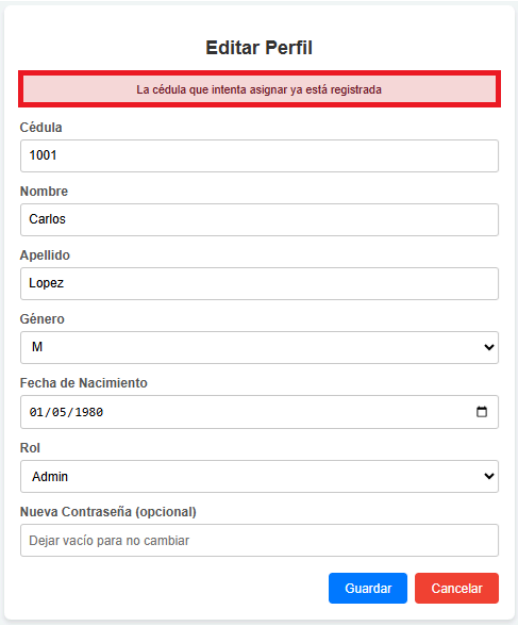


Figura 158, Pantalla de edición de perfil mostrando mensaje de error por documento ya registrado. [Elaboración propia]

Una vez que los datos se editan y guardan con éxito, la plataforma indica un mensaje de confirmación y retorna a la pantalla principal del administrador.

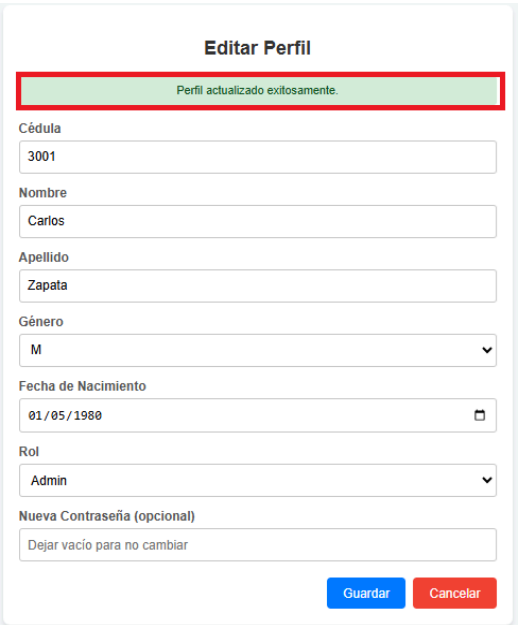


Figura 159, Pantalla de edición del perfil con mensaje de éxito. [Elaboración propia]

En el caso específico de que el cambio incluya la modificación del número de documento o el rol del administrador, y el cambio sea exitoso, una ventana emergente informará que la información de seguridad ha sido modificada y que el administrador debe volver a iniciar sesión para continuar.



Figura 160, Pantalla de edición del perfil con mensaje de éxito y regreso al inicio de sesión. [Elaboración propia]

El tercer y más importante panel se encuentra localizado en el centro y a la derecha de la pantalla. Se caracteriza por ser la tabla de usuarios registrados, funcionando como una lista exhaustiva que muestra la cédula, el nombre, el género y la fecha de nacimiento de todos los usuarios. Esta vista permite al administrador la supervisión total y el mantenimiento de la estructura del sistema.

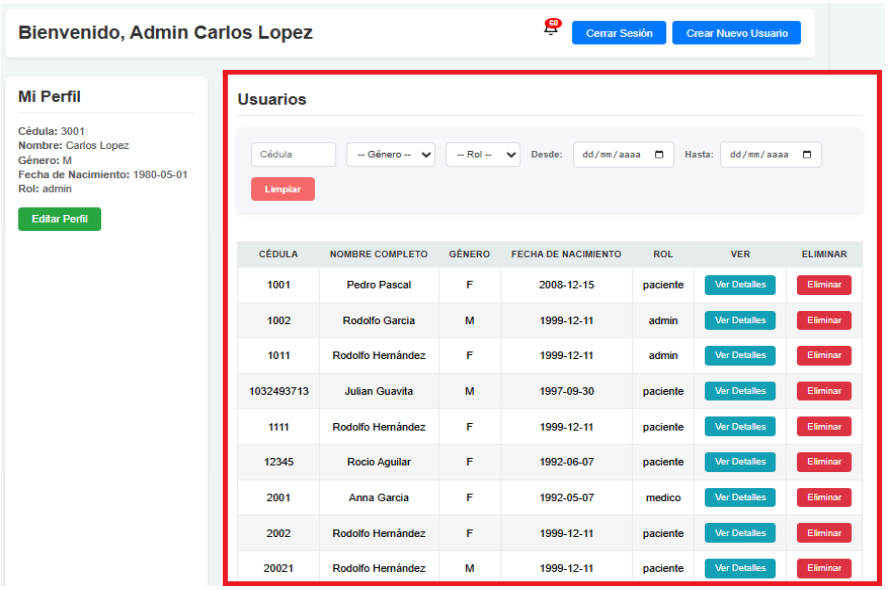


Figura 161, Panel derecho tabla de usuarios. [Elaboración propia]

Para facilitar la búsqueda precisa de usuarios, la tabla está equipada con varios filtros. Las consultas pueden realizarse por género, por rol, por rangos de fecha de nacimiento definiendo una fecha mínima y máxima, o por número de documento. Estos filtros son combinables, y se incluye un botón para eliminar los filtros y restablecer la visualización por orden de cédula ascendente.

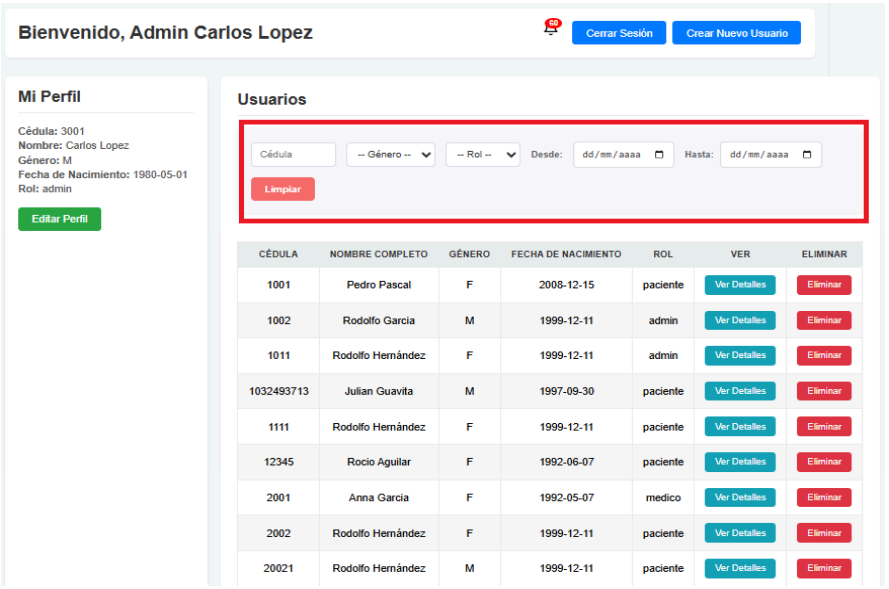


Figura 162, Filtros de la tabla de pacientes. [Elaboración propia]

Adicionalmente, se incluye un botón de acción en la tabla de usuarios para que el administrador pueda eliminar permanentemente a los usuarios. Esta potestad completa en la gestión es esencial para mantener la integridad de la base de datos al permitir la eliminación de cuentas duplicadas, inactivas o creadas por error.

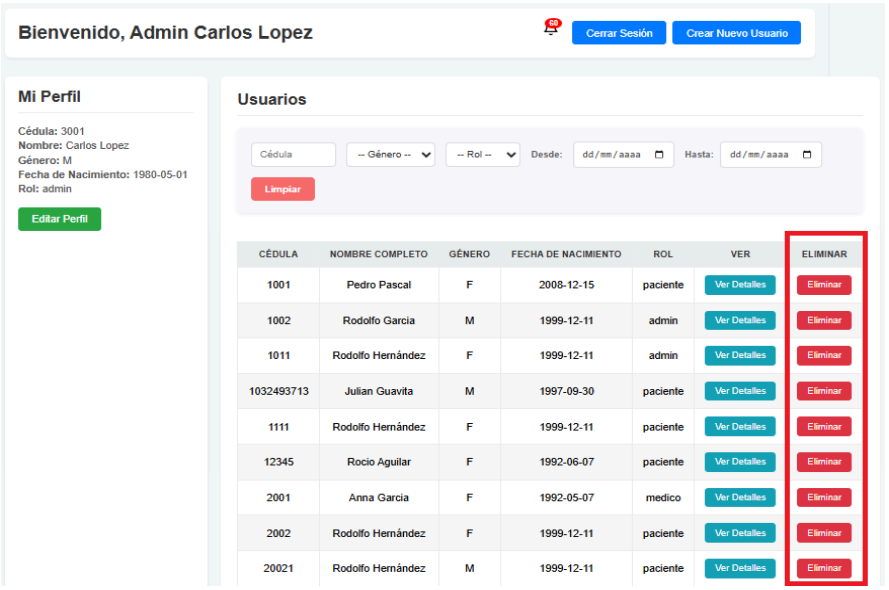


Figura 163, Botón para la eliminación permanente de los usuarios. [Elaboración propia]

Al pulsar el botón de eliminación para un usuario específico, se abre una ventana emergente que solicita la confirmación o cancelación de la eliminación. Este paso de seguridad busca prevenir errores accidentales o eliminaciones no deseadas.

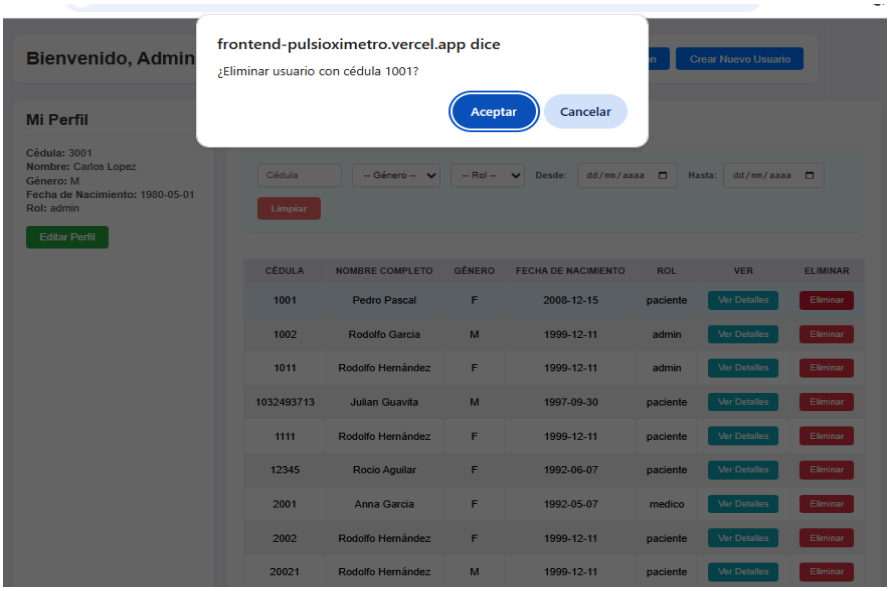


Figura 164, Confirmación para la eliminación permanente de los usuarios. [Elaboración propia]

Finalmente, el registro de cada usuario en la tabla incluye un botón de acción para ver los detalles completos de esa cuenta, lo que dirige a la plataforma a la vista detallada de su historial clínico.

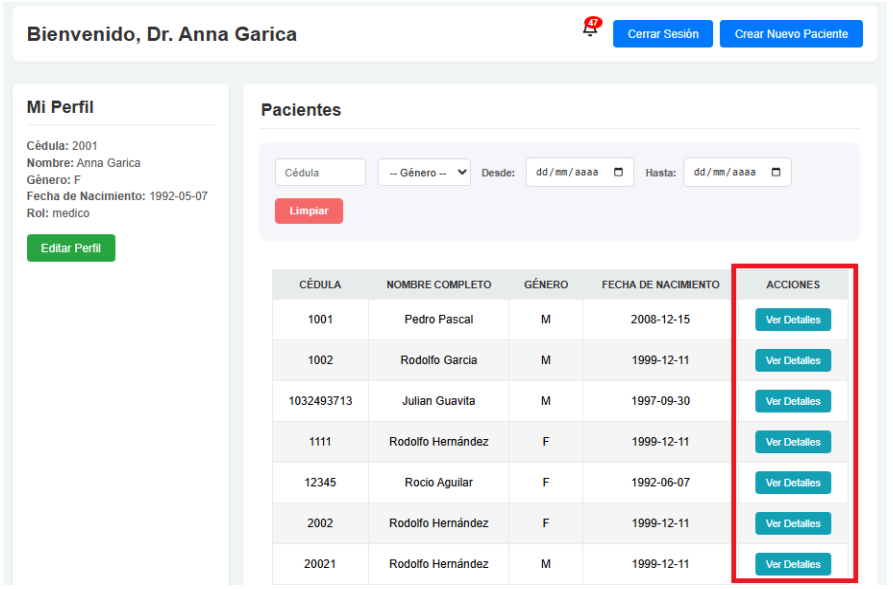


Figura 165, Botón para acceder a la información detallada de los pacientes. [Elaboración propia]

Al usar el botón de ver detalles, la plataforma redirige a la vista detallada del perfil del usuario, organizada en cinco paneles, donde se puede apreciar tanto el perfil básico como el historial clínico.

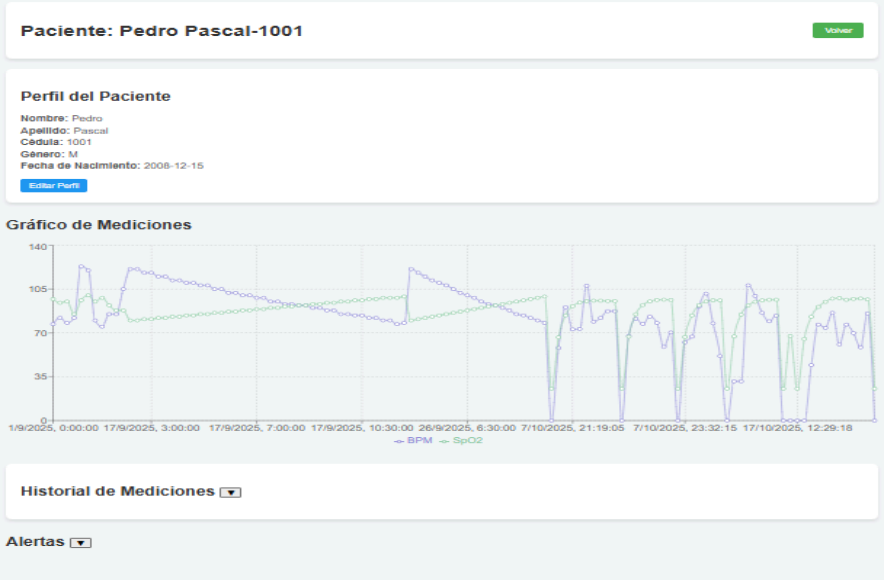


Figura 166, pantalla principal al acceder al botón de la vista de detalles de los pacientes. [Elaboración propia]

El primer componente es un banner superior que proporciona información sobre el paciente, identificándolo por su nombre y número de documento. Este panel incluye un botón que permite volver y retornar a la pantalla principal del administrador.



Figura 167, Panel superior del perfil de los pacientes y botón de regreso. [Elaboración propia]

El segundo panel, ubicado justo después del banner, muestra la información básica del perfil del usuario. Contiene un botón que facilita el acceso a la pantalla de edición del perfil.

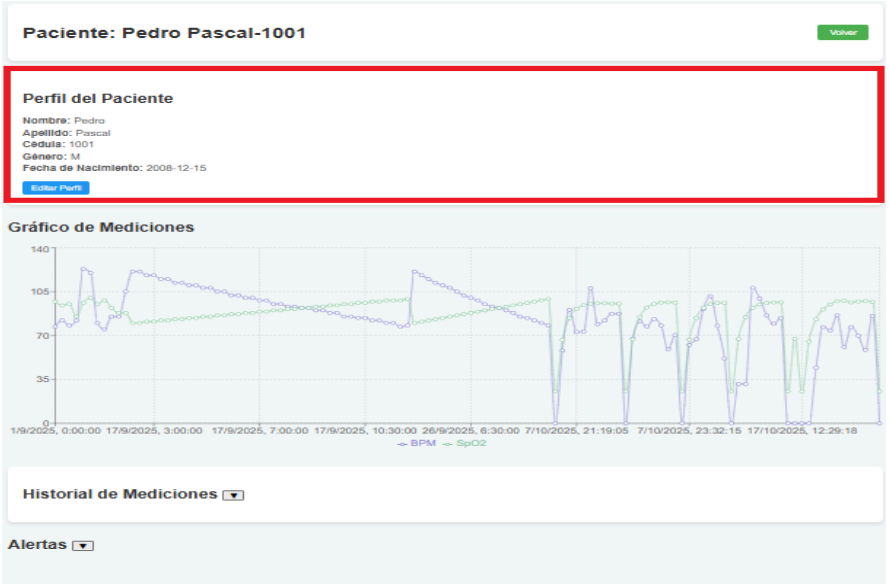


Figura 168, Segundo panel del perfil del paciente. [Elaboración propia]

La pantalla de edición permite al administrador modificar los datos personales del usuario sin excepción. En esta interfaz, se encuentran dos botones de acción: uno para cancelar los cambios y otro para guardarlos y confirmarlos.



Figura 169, Pantalla de cambio de datos de perfil. [Elaboración propia]

Las validaciones críticas del sistema se mantienen: el campo del número de documento solo admite caracteres numéricos, y se previene la duplicidad de documentos.



Figura 170, Pantalla de edición de perfil mostrando mensaje de error por documento ya registrado. [Elaboración propia]

Una vez que los datos se editan y guardan con éxito, la plataforma indica un mensaje de confirmación y retorna a la pantalla principal.

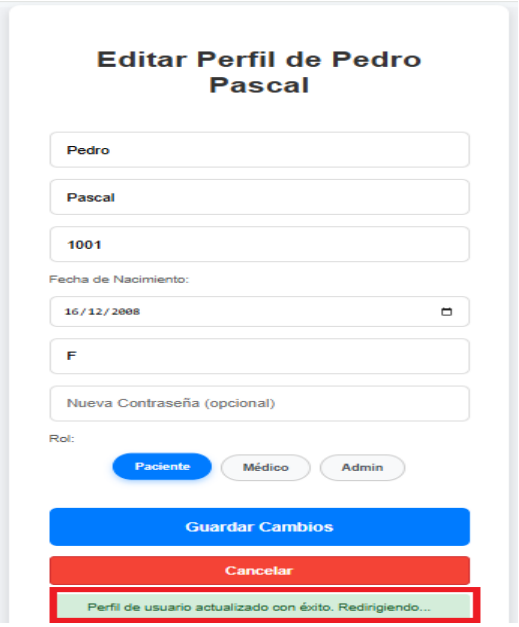


Figura 171, Pantalla de edición del perfil con mensaje de éxito. [Elaboración propia]

En caso de que el cambio incluya el número de documento del usuario y sea exitoso, una ventana emergente informará que la información de seguridad ha sido cambiada y que debe volver a la pantalla principal del administrador.

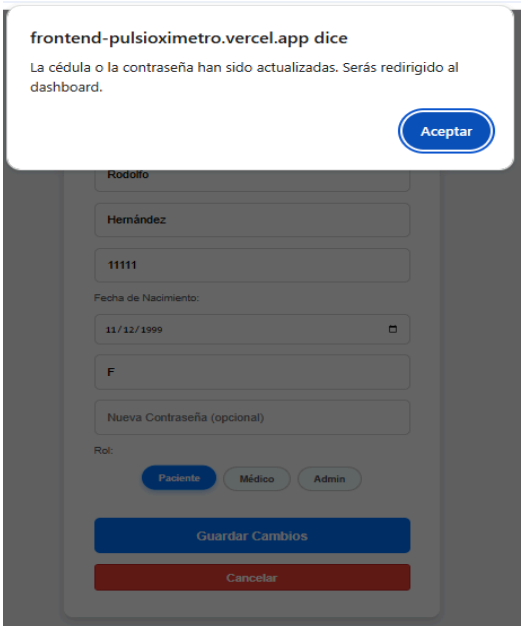


Figura 172, Pantalla de edición del perfil con mensaje de éxito y regreso al inicio de sesión. [Elaboración propia]

El tercer panel se caracteriza por ser una gráfica de tendencia de las mediciones registradas, lo que ayuda al administrador a localizar visualmente problemas en la captación de los signos vitales del paciente en un periodo de tiempo determinado.

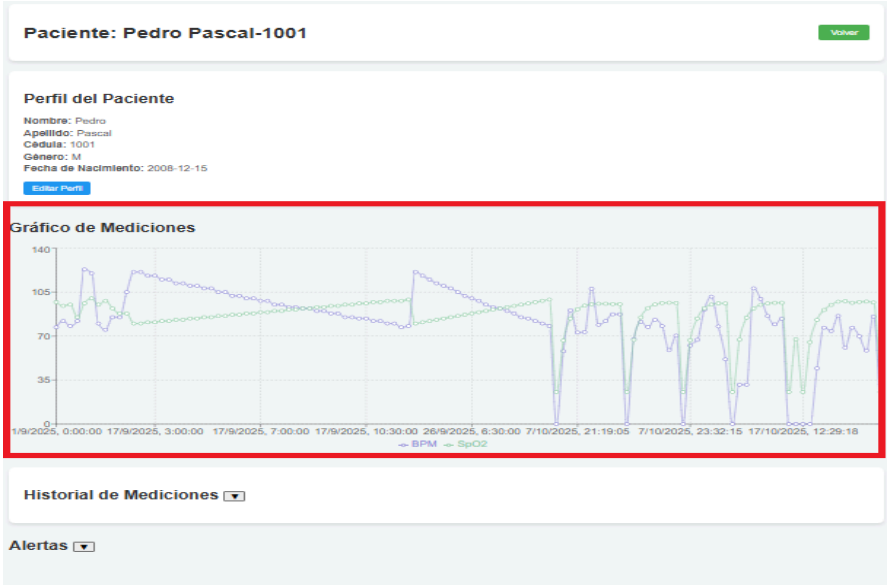


Figura 173, Tercer panel con gráfica de tendencia de datos biométricos. [Elaboración propia]

El cuarto panel, es la tabla de mediciones. Esta tabla almacena el registro histórico de todas las lecturas tomadas con el pulsioxímetro, permitiendo visualizar el ID del registro, los valores de BPM y SpO2, así como la fecha y hora exactas de cada registro.

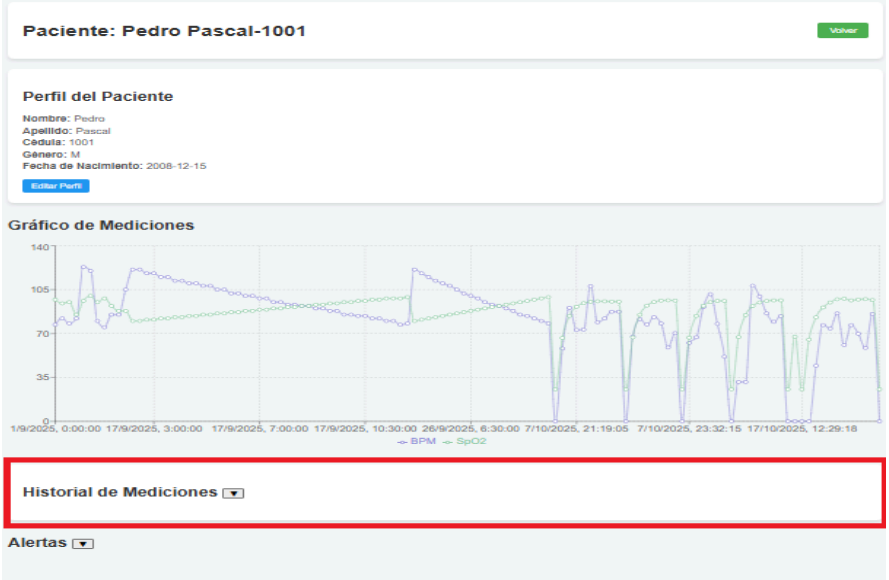


Figura 174, Panel de tabla de registros médicos. [Elaboración propia]

Para facilitar la búsqueda precisa de registros, la tabla incorpora varios filtros. Las consultas pueden realizarse por rangos de fecha definiendo una fecha inicial y final o por rangos de valores estableciendo valores mínimos y máximos para BPM y SpO2. Todos estos filtros son combinables y se incluye un botón para eliminar los filtros.

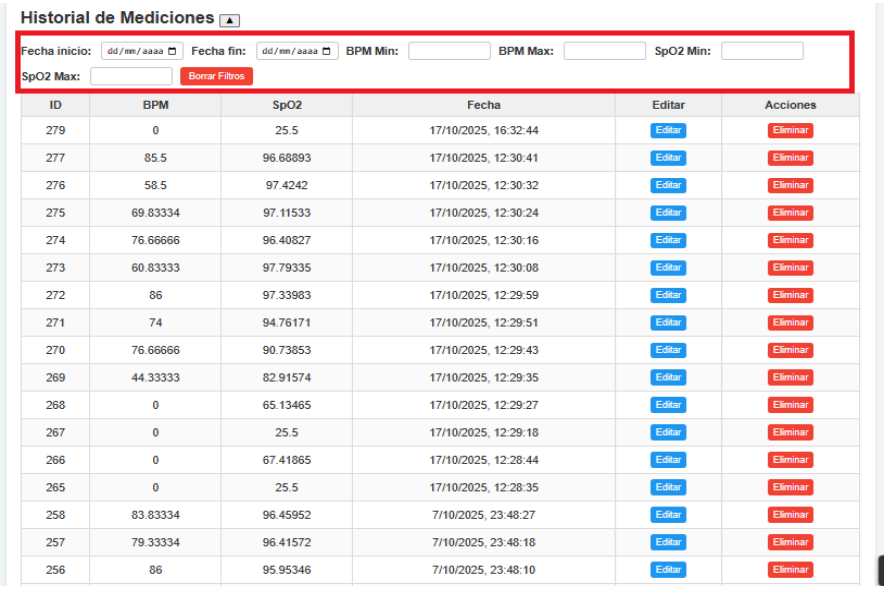


Figura 175, Filtros de la tabla de registros médicos. [Elaboración propia]

Adicionalmente, la tabla incorpora botones de acción para cada registro que permiten la edición de los valores de SpO2 y BPM para corregir pequeños errores que se deban a interferencias o malas lecturas por movimiento.

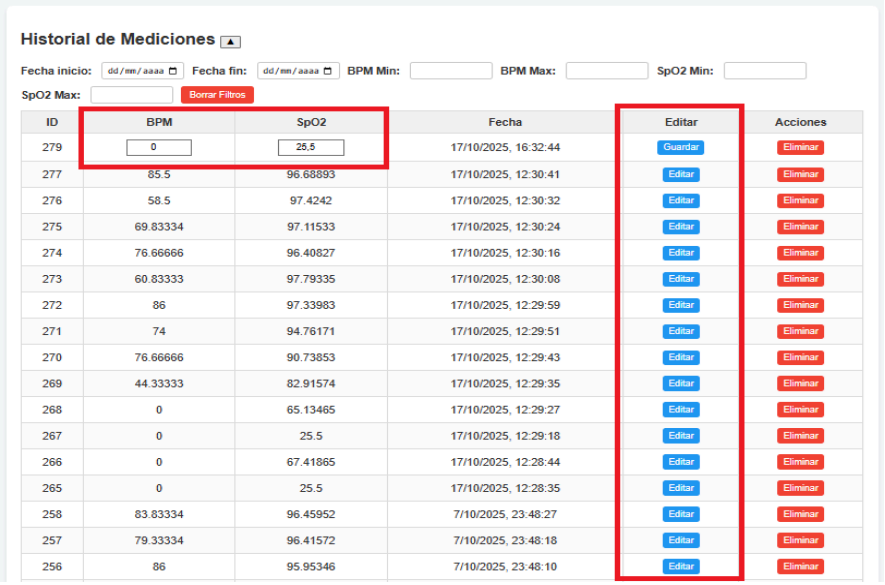


Figura 176, Botón para la edición del spo2 y bpm. [Elaboración propia]

Al final de la tabla, encontramos un segundo botón de acción que le permite al administrador borrar los registros biométricos. Esta función es fundamental para la gestión y limpieza de la base de datos, permitiendo eliminar registros erróneos o irrelevantes.

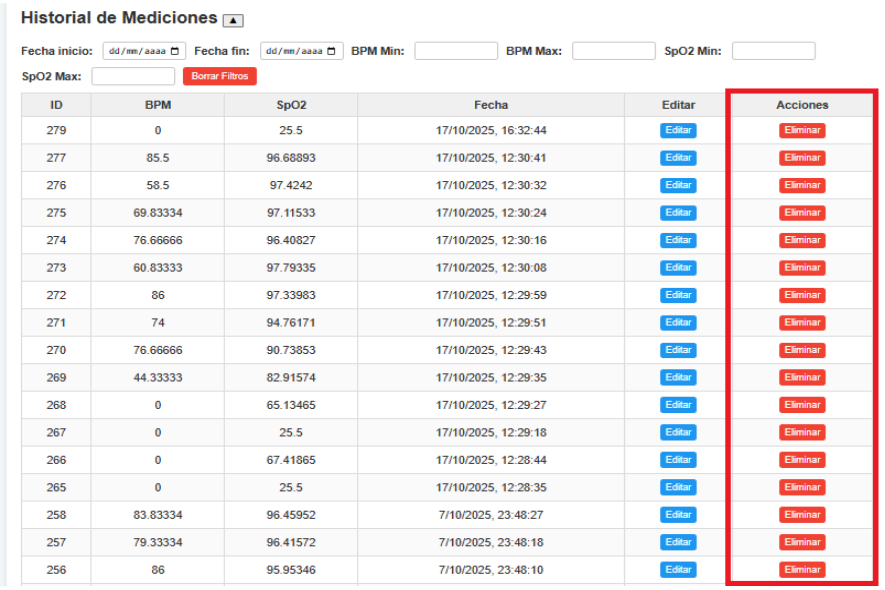


Figura 177, Botón para la eliminación de los registros biométricos. [Elaboración propia]

Al pulsar el botón de eliminación para un registro específico, se abre una ventana emergente para confirmar o cancelar la acción, evitando así errores por pulsaciones accidentales.

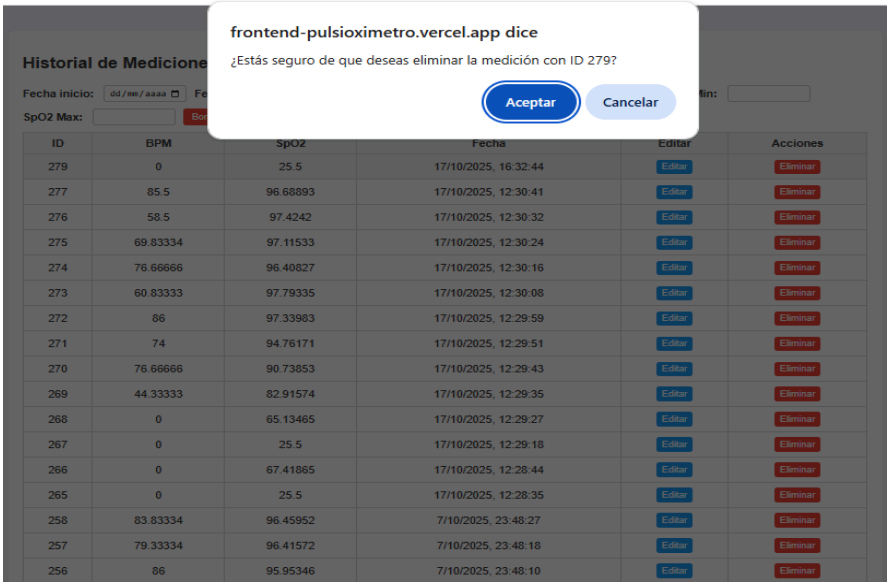


Figura 178, Confirmación para la eliminación permanente de las mediciones. [Elaboración propia]

Para optimizar la visualización y mejorar la fluidez de la pantalla, la tabla cuenta con un botón adicional que le permite contraerse y expandirse, lo que es útil para visualizar el último panel.

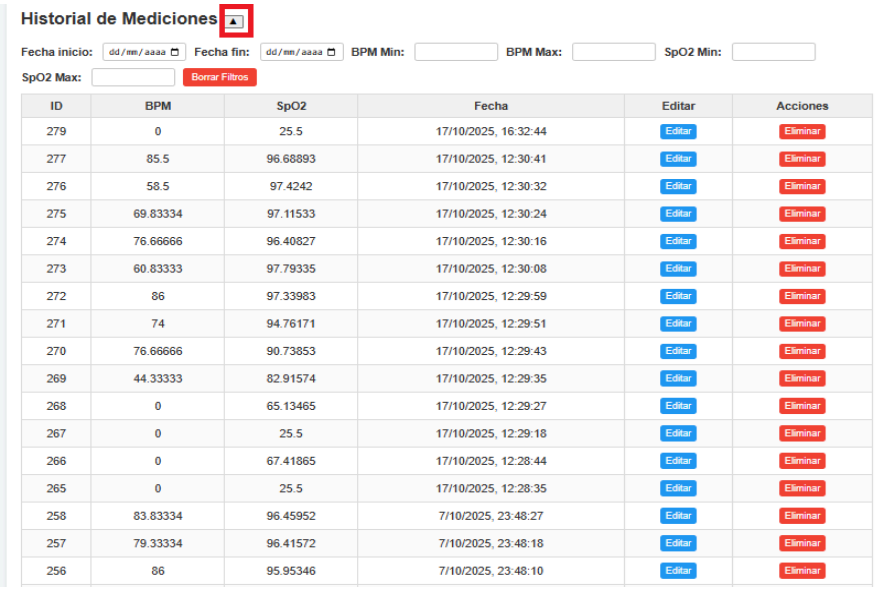


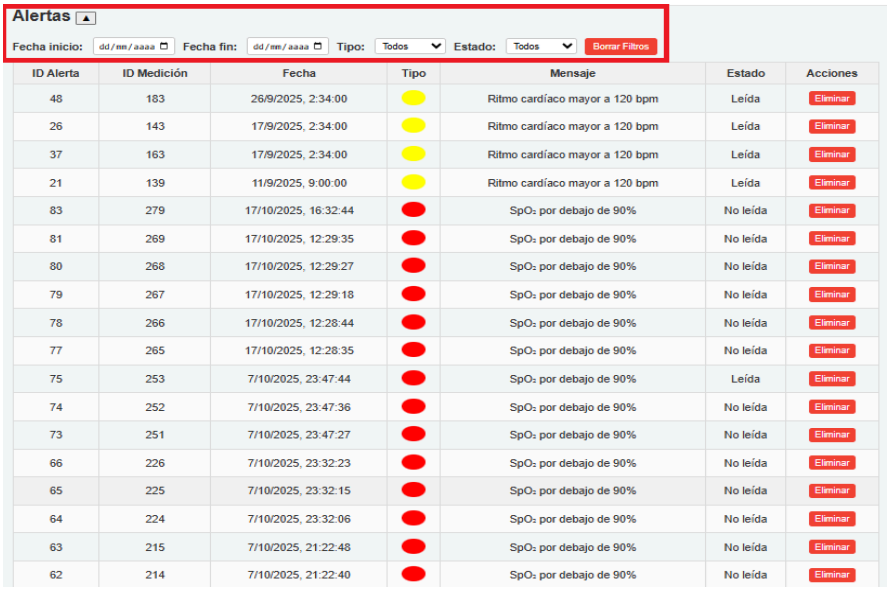
Figura 179, Botón de contracción de la tabla de registros médicos. [Elaboración propia]

Finalmente, el quinto y último panel es la tabla de alertas médicas, ubicada en la parte más inferior de la pantalla. Esta tabla detalla el ID de la alerta, la medición de referencia, la fecha, hora, los datos biométricos, un código de color que identifica el tipo de alerta, un mensaje descriptivo y un estado de lectura que indica el seguimiento por parte del médico.



Figura 180, Panel inferior de tabla de alertas médicas. [Elaboración propia]

De forma análoga, esta tabla incorpora filtros para facilitar la búsqueda, permitiendo realizar consultas por rangos de fechas, por el tipo de alerta y por el estado de lectura. Las consultas permiten aplicar múltiples filtros para obtener resultados precisos, y se incluye un botón para eliminar los filtros.



ID Alerta	ID Medición	Fecha	Tipo	Mensaje	Estado	Acciones
48	183	26/9/2025, 2:34:00		Ritmo cardíaco mayor a 120 bpm	Leída	Eliminar
26	143	17/9/2025, 2:34:00		Ritmo cardíaco mayor a 120 bpm	Leída	Eliminar
37	163	17/9/2025, 2:34:00		Ritmo cardíaco mayor a 120 bpm	Leída	Eliminar
21	139	11/9/2025, 9:00:00		Ritmo cardíaco mayor a 120 bpm	Leída	Eliminar
83	279	17/10/2025, 16:32:44		SpO₂ por debajo de 90%	No leída	Eliminar
81	269	17/10/2025, 12:29:35		SpO₂ por debajo de 90%	No leída	Eliminar
80	268	17/10/2025, 12:29:27		SpO₂ por debajo de 90%	No leída	Eliminar
79	267	17/10/2025, 12:29:18		SpO₂ por debajo de 90%	No leída	Eliminar
78	266	17/10/2025, 12:28:44		SpO₂ por debajo de 90%	No leída	Eliminar
77	265	17/10/2025, 12:28:35		SpO₂ por debajo de 90%	No leída	Eliminar
75	253	7/10/2025, 23:47:44		SpO₂ por debajo de 90%	Leída	Eliminar
74	252	7/10/2025, 23:47:36		SpO₂ por debajo de 90%	No leída	Eliminar
73	251	7/10/2025, 23:47:27		SpO₂ por debajo de 90%	No leída	Eliminar
66	226	7/10/2025, 23:32:23		SpO₂ por debajo de 90%	No leída	Eliminar
65	225	7/10/2025, 23:32:15		SpO₂ por debajo de 90%	No leída	Eliminar
64	224	7/10/2025, 23:32:06		SpO₂ por debajo de 90%	No leída	Eliminar
63	215	7/10/2025, 21:22:48		SpO₂ por debajo de 90%	No leída	Eliminar
62	214	7/10/2025, 21:22:40		SpO₂ por debajo de 90%	No leída	Eliminar

Figura 181, Filtros del panel inferior de la tabla de alertas médicas. [Elaboración propia]

Adicionalmente, la tabla incorpora botones de acción para cada alerta que permiten la eliminación de alertas. Esta función es útil para la limpieza del sistema y para la gestión de alertas antiguas o ya cerradas.

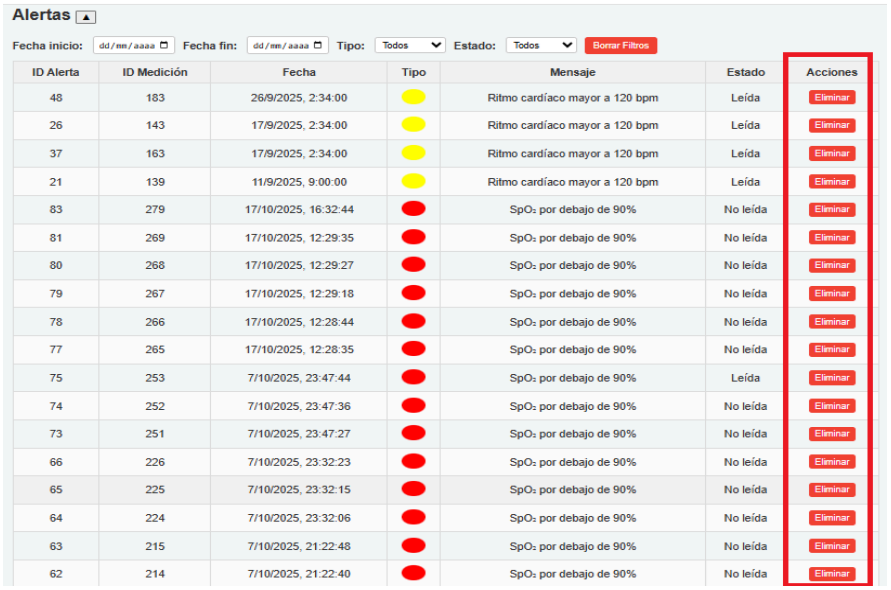


Figura 182, Casilla para la eliminación de alertas. [Elaboración propia]

Al pulsar el botón de eliminación para una alerta específica, se abre una ventana emergente para confirmar o cancelar la eliminación, con el fin de evitar errores por pulsaciones accidentales.

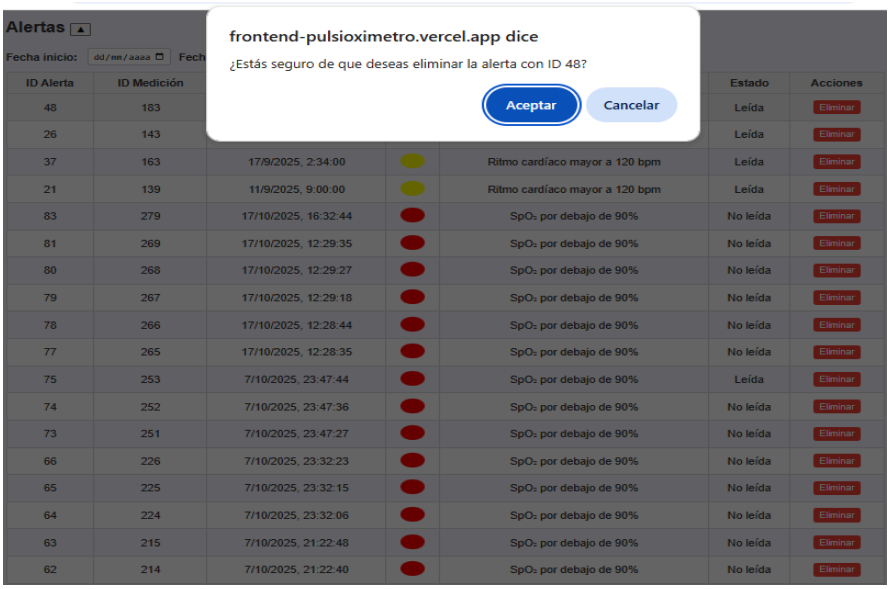


Figura 183, Confirmación para la eliminación permanente de las alertas. [Elaboración propia]

Por último, la tabla de alertas cuenta con un botón que permite contraerse y expandirse con el fin de mejorar la comodidad y la visualización de la pantalla del historial médico del paciente.

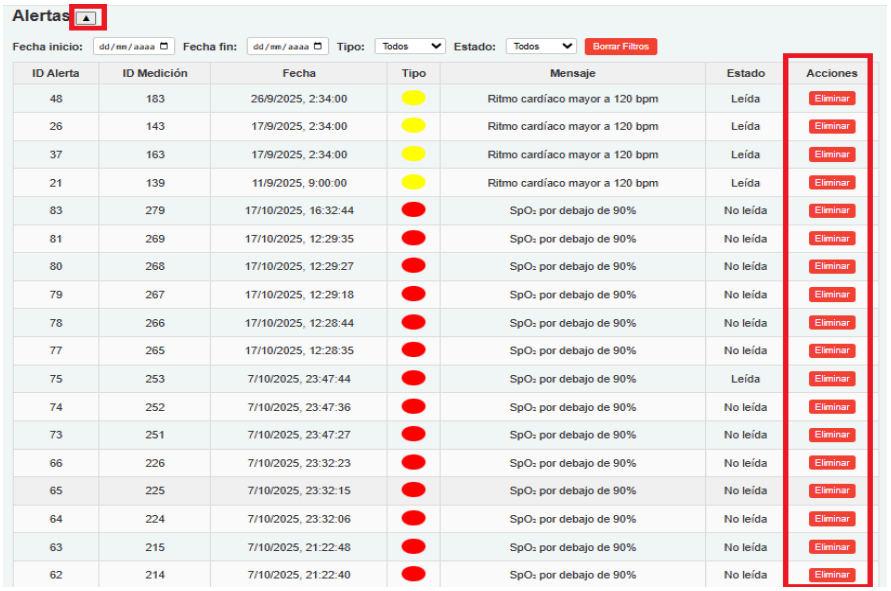


Figura 184, Botón de contracción de la tabla de alertas médicas. [Elaboración propia]

PULSIOXIMETRO

El paso inicial para el uso del pulsioxímetro es el de asegurarse que esté encendido. Para esto, se debe verificar que el *switch* lateral derecha se encuentre en la posición (I), lo cual indica que el dispositivo está energizado y listo para el uso. Si el *switch* está en la posición (O), el dispositivo permanecerá apagado.



Figura 185, Switch de encendido o apagado del interruptor. [Elaboración propia]

Una vez que el interruptor se encuentra en la posición de encendido, se podrá observar que la pantalla se enciende y durante unos segundos nos da un mensaje de bienvenida con el nombre del dispositivo y la versión actual de trabajo.



Figura 186, Pulsador y mensaje de inicio. [Elaboración propia]

Inmediatamente después del mensaje de bienvenida, el sistema preguntará en qué modo operativo se desea emplear el pulsioxímetro: (1) Online o (0) Offline. Para seleccionar cualquiera de estos modos, el usuario debe oprimir la tecla numérica correspondiente en el teclado matricial, lo cual activará la respectiva sesión de medición.



Figura 187, Teclado matricial y selección de modo. [Elaboración propia]

En caso de que el usuario tarde demasiado tiempo en seleccionar el modo de operación, la pantalla indicará por medio de un mensaje que el dispositivo entrará en el modo de bajo consumo. Para reactivar el pulsioxímetro, se debe accionar nuevamente el pulsador de inicio para volver a comenzar el proceso de selección de modo.



Figura 188, Mensaje de modo de bajo consumo y pulsador de inicio. [Elaboración propia]

MODO OFFILINE

Una vez seleccionado el modo Offline, la pantalla mostrará un mensaje indicando al usuario el modo de operación seleccionado y que no es necesaria una conexión a Internet.



Figura 189, Mensaje de inicio del modo offline. [Elaboración propia]

Tras pasar el mensaje anterior, se observa que el sensor MAX30102 se inicializa y su led rojo se enciende. El usuario debe colocar su dedo índice en el sensor del pulsioxímetro. Al hacerlo, la

pantalla OLED comenzará inmediatamente a calibrar los datos del ritmo cardíaco y la saturación de oxígeno del paciente.

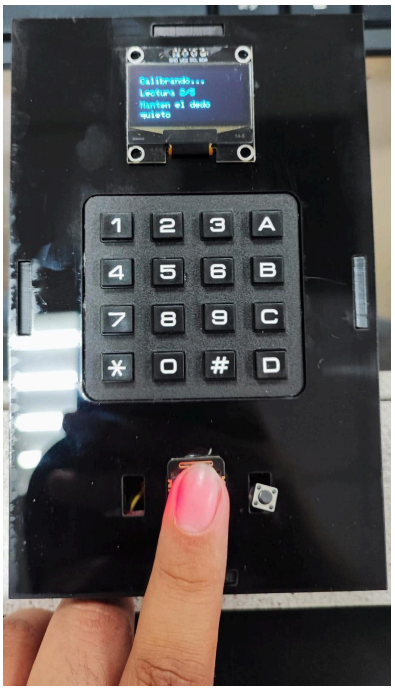


Figura 190, Calibración de datos biométricos. [Elaboración propia]

Una vez que se hayan calibrado los datos biométricos, la pantalla oled comenzará a transmitir inmediatamente los datos del ritmo cardiaco y la saturación de oxígeno del paciente.

Durante las mediciones, el usuario podrá hacer uso del teclado para activar varias funciones específicas:

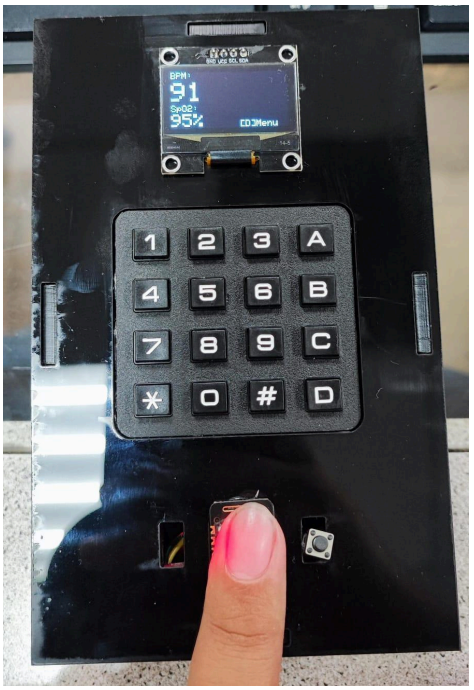


Figura 191, Medición de datos biométricos. [Elaboración propia]

Al oprimir la tecla D, se pausaran las mediciones y se abrirá un submenú que le permitirá al usuario emplear diferentes funciones:



Figura 192, Opciones de selección durante la medición de datos. [Elaboración propia]

Mientras el usuario esté utilizando el dispositivo en modo offline, si retira el dedo del sensor, la pantalla mostrará un mensaje indicando que no se ha detectado un dedo. El usuario puede colocar el dedo nuevamente para continuar con la medición.

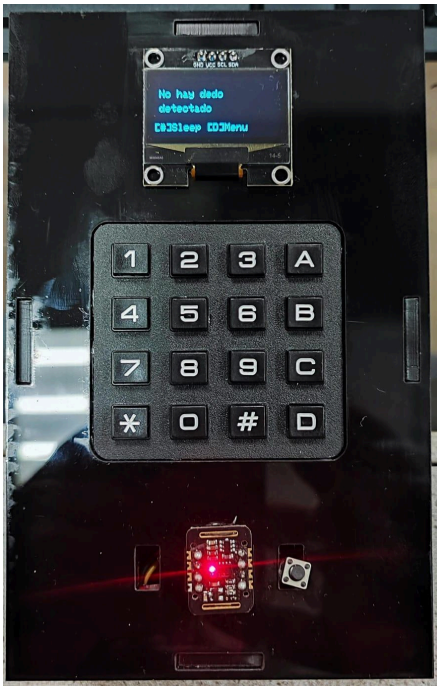


Figura 193, Mensaje de detección del dedo. [Elaboración propia]

Si el usuario sigue sin poner el dedo en el sensor por un tiempo prolongado el dispositivo procederá a mostrar un mensaje donde se indica que se borraron las mediciones y al poner nuevamente el dedo se iniciara una nueva medición omitiendo los registros anteriores

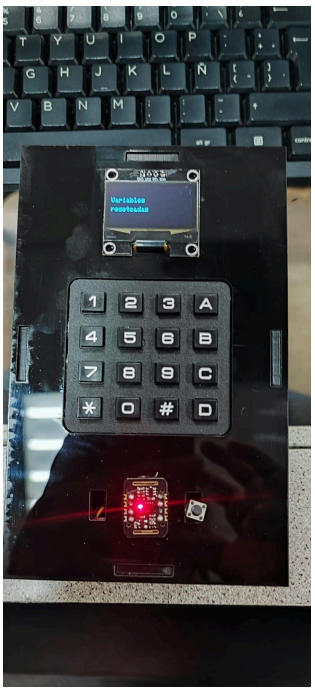


Figura 194, Mensaje de reinicio de mediciones. [Elaboración propia]

En caso de que el usuario pase aún más tiempo sin colocar el dedo en el sensor, el dispositivo mostrará un mensaje indicando que entrará en el modo de bajo consumo. Para usar nuevamente el dispositivo será necesario accionar el pulsador de inicio, ya que de este modo se procede a apagar el dispositivo.



Figura 195, Mensaje de entrada a modo de bajo consumo. [Elaboración propia]

Finalmente, si el usuario acciona el pulsador después de que el dispositivo ha entrado en modo de bajo consumo ya sea por inactividad prolongada o por haber presionado la tecla #, al volver a iniciarse, la pantalla preguntará a con cuál modo de trabajo desea continuar, modo Online y modo Offline. Para seleccionar la opción, el usuario podrá oprimir las teclas 1 o 0 del teclado, respectivamente.



Figura 196, Opciones de reanudación y menú principal.[Elaboración propia]

MODO ONLINE

Si es seleccionado el modo Online, el dispositivo iniciará inmediatamente el proceso de conexión a la red inalámbrica. Si aún no se había establecido una conexión con anterioridad el dispositivo activará el punto de acceso. En este estado, la pantalla indicará al usuario conectarse a la red temporal creada temporalmente y la dirección a la cual acceder para poder configurar el SSID y la contraseña de su red.



Figura 197, Pantalla de Configuración de Wi-Fi. [Elaboración propia]

Si no se completa la configuración en un tiempo de tres minutos, el dispositivo entrará en modo de bajo consumo, teniendo que activar el pulsador para activar el dispositivo nuevamente.



Figura 198, Modo de bajo consumo por inactividad. [Elaboración propia]

Una vez que la conexión se establece con éxito, la pantalla mostrará una confirmación, incluyendo la SSID a la cual se encuentra conectada, así como la dirección IP obtenida dentro de dicha red.



Figura 199, Confirmación de conexión Wi-Fi. [Elaboración propia]

Con la conexión a internet asegurada, el sistema procede a solicitar la identificación del usuario. Recuerde que el usuario deberá ingresar su cédula o identificación, la cual debe tener entre 2 y 10 dígitos. El usuario introducirá su número a través del teclado matricial, pudiendo usar la tecla * para borrar el último dígito o # para confirmar el número de documento



Figura 200, Solicitud y escritura de la identificación. [Elaboración propia]

Si el usuario no ingresa, borra o confirma el ID durante un periodo de 30 segundos, la pantalla mostrará un mensaje reflejando que el tiempo se agotó y el dispositivo realizará automáticamente un

reinicio completo del sistema, forzando la vuelta a la pantalla de bienvenida y a la selección inicial del modo de operación.



Figura 201, Reinicio por inactividad en ingreso de ID. [Elaboración propia]

Una vez que el usuario oprime la tecla # para confirmar el número de documento, el dispositivo mostrará un mensaje de verificación. Si el número de documento no se encuentra registrado, la pantalla indicará que el ID no está registrado e indicará la URL de la página de registro, pasando a otra pantalla de opciones donde el usuario podrá reintentar nuevamente el ID, volver al menú principal y mandar el dispositivo al modo de bajo consumo, con las teclas del teclado A, B y # respectivamente..



Figura 202, Mensaje de ID no registrado y URL para realizar el registro. [Elaboración propia]

Por otro lado, si el número de documento ingresado es válido, la pantalla mostrará un mensaje de confirmación y el dispositivo avanzará automáticamente a la fase de medición. Una vez confirmada la identidad, el usuario debe colocar su dedo índice en el sensor. Al detectar el dedo, la pantalla OLED comenzará a mostrar el proceso de calibración del ritmo cardíaco y la saturación de oxígeno, al igual que en el modo offline. La información será enviada continuamente a la base de datos.



Figura 203, Inicio de medición y transmisión de datos. [Elaboración propia]

Durante las mediciones, el usuario podrá hacer uso del teclado para activar seis funciones específicas al ingresar al menú presionando la tecla D:

- Para ver la información como el nombre de la red a la cual se está conectado, el ID ingresado y la calidad de la señal, debe presionar la tecla 1
- Para resetear las variables, debe presionar la tecla 2
- Para cambiar de modo, presione la tecla 3
- Para resetear la información referente a la conexión Wifi, presione la tecla 4.
- De igual modo tendrá la opción del modo de bajo consumo con la tecla # o volver a la pantalla de mediciones con la tecla D.



Figura 204, Opciones de selección durante la medición de datos. [Elaboración propia]

Mientras el usuario esté utilizando el dispositivo en modo *online*, si retira el dedo del sensor, la pantalla mostrará un mensaje indicando que no se ha detectado un dedo, tal como sucede en el modo offline. La medición y la transmisión de datos se detienen, pero el ID del paciente se mantiene cargado. El usuario puede colocar el dedo nuevamente para continuar con la medición.



Figura 205, Mensaje de no detección del dedo. [Elaboración propia]

Si el usuario sigue sin poner el dedo en el sensor por un tiempo prolongado, el dispositivo procederá a mostrar un mensaje donde se indica que se borraron las mediciones y, al poner nuevamente el dedo, se iniciará una nueva medición omitiendo los registros anteriores.



Figura 206, Mensaje de reinicio de mediciones. [Elaboración propia]

En caso de que el usuario pase aún más tiempo sin colocar el dedo en el sensor, el dispositivo mostrará un mensaje indicando que entrará en el modo de bajo consumo. Para usar nuevamente el dispositivo será necesario accionar el pulsador de inicio, ya que de este modo se procede a apagar el dispositivo.



Figura 207, Mensaje de entrada a modo de bajo consumo. [Elaboración propia]

Finalmente, una vez que el dispositivo entra en modo de bajo consumo por inactividad o por haber pulsado la tecla # durante la medición, el usuario deberá accionar el pulsador para activarlo. Al

despertar, el sistema no reanuda la medición de inmediato, sino que presenta un menú de opciones para seleccionar el modo de trabajo.



Figura 208, Opciones de reanudación y menú principal. [Elaboración propia]

Al optar por continuar con el modo online, el sistema activa un protocolo de conexión automática a la red previamente configurada. La pantalla pasará entonces a solicitar al usuario ingresar su identificación.



Figura 209, Protocolo de autoconexión Wifi. [Elaboración propia]

En todos los casos al entrar al modo de bajo consumo, la reactivación será mediante el pulsador, el cual nos llevará al menú de selección de modo, al volver a iniciarse, la pantalla preguntará a con cuál

modo de trabajo desea continuar, modo Online y modo Offline. Para seleccionar la opción, el usuario podrá oprimir las teclas 1 o 0 del teclado, respectivamente.



Figura 210, Opciones de reanudación y menú principal. [Elaboración propia

A.2 CÓDIGO PARA EL PROGRAMA OPERACIONAL DEL PULSIOXÍMETRO

En este apartado se presenta el código fuente completo del pulsioxímetro. Dicho código incluye la lógica detallada para la gestión de diversas funcionalidades y componentes, que se resumen a continuación:

- Cálculo de Datos Biométricos: Algoritmos implementados para el procesamiento de señales y la determinación de los valores de frecuencia cardíaca (BPM) y saturación de oxígeno (SpO2).
- Conexión y Comunicación: Módulos para establecer la conexión Wi-Fi y gestionar la comunicación segura entre el dispositivo y la base de datos, permitiendo el almacenamiento y la recuperación de la información.
- Manejo de Periféricos: Rutinas de software específicas para la interacción y control de los componentes de entrada y salida, incluyendo el teclado matricial 4x4 y la pantalla OLED.
- Integración de Procesos: Estructura general del código diseñada para la integración eficiente de todas las funciones mencionadas, garantizando una operación fluida y una experiencia de usuario intuitiva.

```
#include <Wire.h>
#include "MAX30105.h"
#include "heartRate.h"
#include <esp32-hal.h>
#include <WiFiManager.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SH1106_ESP32.h>
#include <HTTPClient.h>
#include <Keypad.h>
#include <ArduinoJson.h>

//=====================================================================
======
// CONFIGURACIÓN DE SUPABASE
//=====================================================================
======
const char* SUPABASE_URL = "https://aeuliysglbtusyqkayuc.supabase.co/rest/v1/mediciones";
const char* SUPABASE_USUARIOS_URL =
"https://aeuliysglbtusyqkayuc.supabase.co/rest/v1/usuarios";
const char* SUPABASE_KEY =
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsInJlZiI6ImFldWxpeXNn
bGJ0dXN5cWtheXVjIiwicm9sZSI6InNlcnZpY2Vfcm9sZSIsImlhdCI6MTc1NzcyMjU0OSwiZXhwIj
oyMDczMjk4NTQ5fQ.YgItMu7MNSiQE46gvXviQ3x1jE1kk8y-TEVgHL48b5g";
const char* URL_REGISTRO = "frontend-pulsioximetro.vercel.app";

//=====================================================================
======
// CONFIGURACIÓN DEL TECLADO MATRICIAL 4x4
//=====================================================================
======
const byte ROWS = 4;
const byte COLS = 4;

byte rowPins[ROWS] = {13, 12, 14, 27};
byte colPins[COLS] = {26, 25, 33, 32};

char keys[ROWS][COLS] = {
  { '1', '4', '7', '*' },
  { '2', '5', '8', '0' },
  { '3', '6', '9', '#' },
  { 'A', 'B', 'C', 'D' }
};

Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

//=====================================================================
======
// TIEMPOS (en milisegundos)
```

```
//================================================================================
======
const unsigned long WELCOME_DELAY = 3000;
const unsigned long MODE_SELECT_TIMEOUT = 30000;        // 30 seg
const unsigned long ID_INPUT_TIMEOUT = 30000;           // 30 seg
const unsigned long MODE_DISPLAY_DELAY = 1000;
const unsigned long WIFI_AUTOCONNECT_TIMEOUT = 15000;   // 15 seg autoconnect
const unsigned long WIFI_CONFIG_TIMEOUT = 180000;       // 3 min portal cautivo
const unsigned long WIFI_RECONNECT_INTERVAL = 30000;    // 30 seg entre intentos de
reconexión
const unsigned long WIFI_CHECK_INTERVAL = 60000;        // 60 seg entre revisiones de WiFi
const unsigned long LOW_POWER_INFO_DELAY = 3000;
const unsigned long NO_FINGER_PRINT_INTERVAL = 5000;
const unsigned long RESET_VARS_TIMEOUT = 30000;         // 30 seg
const unsigned long LOW_POWER_TIMEOUT = 120000;         // 2 min
const unsigned long WAKE_DECISION_TIMEOUT = 30000;      // 30 seg (aumentado para tener
más tiempo)
const unsigned long DATA_SEND_INTERVAL = 5000;          // 5 seg
const unsigned long MENU_TIMEOUT = 30000;               // 30 seg timeout universal

//================================================================================
======
// CONFIGURACIÓN DE VALIDACIÓN DE ID
//================================================================================
======
const int MIN_ID_LENGTH = 2;  // ⬅️ Cambiar a 10 en producción
const int MAX_ID_LENGTH = 10;

String idPacienteActual = "OFFLINE_USER";

//================================================================================
======
// CONFIGURACIÓN DEL SENSOR MAX30102
//================================================================================
======
MAX30105 particleSensor;
#define RATE_SIZE 6
#define FINGER_ON 35000

byte rates[RATE_SIZE];
byte rateSpot = 0;
long lastBeat = 0;
float beatAvg = 0;

double avered = 0, aveir = 0;
double sumirrms = 0, sumredrms = 0;
int i = 0;
const int Num = 100;
```

```
float ESpO2 = 0;
const float FSpO2 = 0.7;
const double frate = 0.95;
float smoothedR = 1.0;
const float alpha = 0.7;

// ⬅️ NUEVO: Variables para mejorar estabilidad de SpO2
int validReadings = 0;          // Contador de lecturas válidas
const int MIN_VALID_READINGS = 5; // Mínimo de lecturas antes de mostrar
bool sensorWarmedUp = false;      // Sensor necesita calentarse
unsigned long sensorStartTime = 0; // Tiempo de inicio del sensor

//=============================================================================
======
// VARIABLES DE CONTROL
//=============================================================================
======
unsigned long lastDetectedTime = 0;
unsigned long lastPrintTime = 0;
unsigned long lastSendTime = 0;
unsigned long programStartTime = 0;
unsigned long lastActivityTime = 0;
unsigned long lastSignalCheckTime = 0;
unsigned long lastWiFiReconnectAttempt = 0;  // ⬅️ NUEVO: Para reconexión automática
unsigned long lastWiFiCheck = 0;             // ⬅️ NUEVO: Para revisar WiFi periódicamente

bool isFingerDetected = false;
bool lowPowerMode = false;
bool isOnline = false;
bool lastDataSent = false;
bool modeSelected = false;
bool waitingForWakeDecision = false;

//=============================================================================
======
// PINES Y HARDWARE
//=============================================================================
======
#define BUTTON_PIN 15
#define LED_PIN 2
#define SCREEN_ADDR 0x3C

Adafruit_SH1106_ESP32 display;

//=============================================================================
======
// INTERRUPCIÓN PARA DESPERTAR
```

```
//=============================================================================
======
void IRAM_ATTR handleWakeUp() {
  // Solo despierta
}

//=============================================================================
======
// ACTUALIZAR TIEMPO DE ACTIVIDAD
//=============================================================================
======
void updateActivity() {
  lastActivityTime = millis();
}

//=============================================================================
======
// VERIFICAR TIMEOUT Y ENTRAR EN SLEEP SI ES NECESARIO
//=============================================================================
======
bool checkMenuTimeout(unsigned long timeout) {
  if (millis() - lastActivityTime > timeout) {
    Serial.println("⏰ Timeout detectado - Entrando en sleep");
    display.clearDisplay();
    display.setCursor(0, 20);
    display.println("Timeout");
    display.setCursor(0, 35);
    display.println("Entrando en sleep...");
    display.display();
    delay(2000);
    enterLowPowerModeNoMode();
    return true;
  }
  return false;
}

//=============================================================================
======
// RECONEXIÓN AUTOMÁTICA WiFi (NUEVA FUNCIÓN - OPTIMIZADA)
//=============================================================================
======
bool autoReconnectWiFi() {
  // Solo intentar si ha pasado suficiente tiempo desde el último intento
  if (millis() - lastWiFiReconnectAttempt < WIFI_RECONNECT_INTERVAL) {
    return false;
  }

  lastWiFiReconnectAttempt = millis();
```

```
Serial.println("🔄 Intentando reconexión WiFi automática...");

// NO mostrar en pantalla si hay dedo detectado (priorizar mediciones)
if (!isFingerDetected) {
  display.clearDisplay();
  display.setCursor(0, 15);
  display.println("WiFi desconectado");
  display.setCursor(0, 30);
  display.println("Reconectando...");
  display.display();
}

WiFi.reconnect();

// Esperar solo 5 segundos para reconexión (REDUCIDO para no afectar mediciones)
unsigned long startAttempt = millis();
while (WiFi.status() != WL_CONNECTED && millis() - startAttempt < 5000) {
  delay(100);  // Delay más corto
  Serial.print(".");
}
Serial.println();

if (WiFi.status() == WL_CONNECTED) {
  Serial.println("✅ WiFi reconectado exitosamente");
  Serial.print("📶 IP: ");
  Serial.println(WiFi.localIP());

  // ⬅️ Encender LED al reconectar
  digitalWrite(LED_PIN, HIGH);

  // Solo mostrar mensaje si NO hay dedo
  if (!isFingerDetected) {
    display.clearDisplay();
    display.setCursor(0, 20);
    display.println("WiFi reconectado!");
    display.display();
    delay(1000);
  }

  return true;
} else {
  Serial.println("❌ Reconexión WiFi falló - Continuando sin conexión");

  // ⬅️ Apagar LED si no hay conexión
  digitalWrite(LED_PIN, LOW);

  return false;
```

```
  }
}

//=====================================================================
======
// VERIFICAR ESTADO WiFi Y RECONECTAR SI ES NECESARIO (OPTIMIZADA)
//=====================================================================
======
void checkAndMaintainWiFi() {
  if (!isOnline) {
    return;  // No hacer nada si estamos en modo offline
  }

  // Solo revisar cada 60 segundos (no en cada loop)
  if (millis() - lastWiFiCheck < WIFI_CHECK_INTERVAL) {
    return;
  }

  lastWiFiCheck = millis();

  if (WiFi.status() != WL_CONNECTED) {
    Serial.println("⚠️ WiFi desconectado detectado");

    // IMPORTANTE: Solo reconectar si NO hay dedo detectado
    // Esto evita interrumpir las mediciones activas
    if (!isFingerDetected) {
      autoReconnectWiFi();
    } else {
      Serial.println("⏸️  Reconexión pospuesta - Medición en progreso");
    }
  }
}

//=====================================================================
======
// VERIFICAR ID EN SUPABASE
//=====================================================================
======
bool verificarIDenSupabase(String cedula) {
  if(WiFi.status() != WL_CONNECTED) {
    Serial.println("❌ Error: WiFi no conectado para verificación");
    return false;
  }

  // Verificar señal WiFi
  int rssi = WiFi.RSSI();
  Serial.print("📶 Señal WiFi: ");
  Serial.print(rssi);
```

```
Serial.println(" dBm");

HTTPClient http;

// Construir URL de consulta
String url = String(SUPABASE_USUARIOS_URL) + "?cedula=eq." + cedula + "&select=cedula";

Serial.println("🔍 Verificando ID en base de datos...");
Serial.println("URL: " + url);

http.begin(url);

// Ajustar timeout según señal
if (rssi > -70) {
  http.setTimeout(3000);  // Señal buena: 3 segundos
} else {
  http.setTimeout(8000);  // Señal débil: 8 segundos
  Serial.println("⚠️ Señal débil - Usando timeout extendido");
}

http.addHeader("apikey", SUPABASE_KEY);
http.addHeader("Authorization", String("Bearer ") + SUPABASE_KEY);

unsigned long startTime = millis();
int httpResponseCode = http.GET();
unsigned long duration = millis() - startTime;

Serial.print("⏱️ Verificación duró: ");
Serial.print(duration);
Serial.println("ms");

if (httpResponseCode > 0) {
  String response = http.getString();
  Serial.println("📥 Response Code: " + String(httpResponseCode));
  Serial.println("📥 Response: " + response);

  http.end();

  // Si la respuesta es "[]" significa que no existe
  // Si la respuesta contiene datos, existe
  if (httpResponseCode == 200) {
    if (response == "[]" || response.length() < 3) {
      Serial.println("❌ ID no encontrado en base de datos");
      return false;
    } else {
      Serial.println("✅ ID verificado exitosamente");
      return true;
    }
```

```
    } else {
      Serial.println("⚠️ Código de respuesta inesperado");
      return false;
    }
  } else {
    Serial.print("❌ Error HTTP en verificación: ");
    Serial.print(httpResponseCode);
    Serial.print(" (duró ");
    Serial.print(duration);
    Serial.println("ms)");
    http.end();
    return false;
  }
}

//=============================================================================
======
// VERIFICAR ID EN SUPABASE CON OPCIÓN DE CANCELAR
//=============================================================================
======
// Retorna: 1 = ID verificado, 0 = ID no encontrado, -1 = Cancelado por usuario
int verificarIDenSupabaseConCancelacion(String cedula) {
  if (WiFi.status() != WL_CONNECTED) {
    Serial.println("❌ WiFi no conectado");
    return 0;
  }

  int rssi = WiFi.RSSI();
  Serial.print("📶 Señal WiFi: ");
  Serial.print(rssi);
  Serial.println(" dBm");

  HTTPClient http;

  // Construir URL de consulta
  String url = String(SUPABASE_USUARIOS_URL) + "?cedula=eq." + cedula + "&select=cedula";

  Serial.println("🔍 Verificando ID en base de datos...");
  Serial.println("URL: " + url);

  http.begin(url);

  // Timeout más largo para dar tiempo a cancelar
  http.setTimeout(15000);  // 15 segundos

  http.addHeader("apikey", SUPABASE_KEY);
  http.addHeader("Authorization", String("Bearer ") + SUPABASE_KEY);
```

```
unsigned long startTime = millis();
unsigned long lastDisplayUpdate = millis();
int dotCount = 0;

// Iniciar la petición en background
int httpResponseCode = -1;
bool requestStarted = false;

// Loop para permitir cancelación mientras se espera
while (true) {
  // Actualizar indicador visual cada 500ms
  if (millis() - lastDisplayUpdate > 500) {
    display.clearDisplay();
    display.setTextSize(1);
    display.setCursor(0, 5);
    display.println("Verificando ID...");
    display.setCursor(0, 20);
    display.setTextSize(2);
    display.println(cedula);
    display.setTextSize(1);
    display.setCursor(0, 40);

    // Indicador de progreso (puntos animados)
    String dots = "Conectando";
    for (int i = 0; i < dotCount; i++) {
      dots += ".";
    }
    display.println(dots);
    dotCount = (dotCount + 1) % 4;

    display.setCursor(0, 55);
    display.println("[D] Cancelar");
    display.display();

    lastDisplayUpdate = millis();
  }

  // Hacer la petición HTTP una sola vez
  if (!requestStarted) {
    httpResponseCode = http.GET();
    requestStarted = true;
    unsigned long duration = millis() - startTime;

    Serial.print("⏱️ Verificación duró: ");
    Serial.print(duration);
    Serial.println("ms");

    // Procesar resultado
```

```
    if (httpResponseCode > 0) {
      String response = http.getString();
      Serial.println("📥 Response Code: " + String(httpResponseCode));
      Serial.println("📥 Response: " + response);

      http.end();

      if (httpResponseCode == 200) {
        if (response == "[]" || response.length() < 3) {
          Serial.println("❌ ID no encontrado en base de datos");
          return 0;  // ID no encontrado
        } else {
          Serial.println("✅ ID verificado exitosamente");
          return 1;  // ID verificado
        }
      } else {
        Serial.println("⚠️ Código de respuesta inesperado");
        http.end();
        return 0;  // Error, tratar como no encontrado
      }
    } else {
      Serial.print("❌ Error HTTP en verificación: ");
      Serial.println(httpResponseCode);
      http.end();
      return 0;  // Error de conexión
    }
  }

  // Verificar si usuario presiona [D] para cancelar
  char key = keypad.getKey();
  if (key == 'D') {
    Serial.println("\n🚫 Usuario canceló la verificación");
    http.end();
    return -1;  // Cancelado por usuario
  }

  delay(50);  // Pequeña pausa para no saturar el CPU
 }
}

//================================================================================
// MOSTRAR PANTALLA NO REGISTRADO
//================================================================================
void mostrarPantallaNoRegistrado(String cedula) {
 // Primera pantalla: Información de registro
 display.clearDisplay();
```

```
display.setTextSize(1);
display.setCursor(0, 0);
display.println("ID no registrado");
display.setCursor(0, 15);
display.println("Cedula: " + cedula);
display.setCursor(0, 30);
display.println("Registrate en:");
display.setCursor(0, 45);
display.println(URL_REGISTRO);
display.display();

Serial.println("╔═══════════════════════════════════════╗");
Serial.println("║  ID NO REGISTRADO          ║");
Serial.println("╚═══════════════════════════════════════╝");
Serial.println("Cédula ingresada: " + cedula);
Serial.println("Por favor regístrate en: " + String(URL_REGISTRO));
Serial.println("");
Serial.println("Si necesitas ayuda, contacta al administrador del sistema.");

// Esperar 5 segundos para que lean el mensaje
delay(5000);

// Segunda pantalla: Opciones
display.clearDisplay();
display.setTextSize(1);
display.setCursor(0, 0);
display.println("Que deseas hacer?");
display.setCursor(0, 15);
display.println("[A] Reintentar ID");
display.setCursor(0, 28);
display.println("[B] Menu Principal");
display.setCursor(0, 41);
display.println("[#] Sleep");
display.setCursor(0, 54);
display.println("Solicita ayuda si");
display.setCursor(0, 62);
display.println("es necesario");
display.display();

Serial.println("━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━");
Serial.println("Opciones disponibles:");
Serial.println("🔄 [A] Reintentar ingreso de ID (modo Online)");
Serial.println("🏠 [B] Volver al menú principal");
Serial.println("💤 [#] Entrar en modo Sleep");
Serial.println("━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━");

// Esperar a que presione una opción
while (true) {
```

```
    char key = keypad.getKey();
    if (key == 'A') {
      Serial.println("\n🔄 Reintentando ingreso de ID...");
      display.clearDisplay();
      display.setCursor(0, 25);
      display.println("Reintentando...");
      display.display();
      delay(1000);
      return;  // Retornar para volver a pedir el ID
    }
    if (key == 'B') {
      Serial.println("\n🏠 Volviendo al menú principal...");
      display.clearDisplay();
      display.setCursor(0, 20);
      display.println("Volviendo al");
      display.setCursor(0, 35);
      display.println("menu principal...");
      display.display();
      delay(1500);
      // Resetear variables y volver a setup
      modeSelected = false;
      isOnline = false;
      idPacienteActual = "OFFLINE_USER";
      setup();  // Volver al inicio
      return;
    }
    if (key == '#') {
      Serial.println("\n💤 Entrando en modo Sleep...");
      display.clearDisplay();
      display.setCursor(0, 20);
      display.println("Entrando en");
      display.setCursor(0, 35);
      display.println("modo Sleep...");
      display.display();
      delay(2000);
      // Entrar en sleep sin seleccionar modo
      enterLowPowerModeNoMode();
      return;
    }
    delay(100);
  }
}

//================================================================================
// SOLICITAR ID DEL PACIENTE CON VERIFICACIÓN
//================================================================================
```

```
String solicitarIDPaciente() {
  String idTemp = "";
  unsigned long startTime = millis();
  bool idVerificado = false;

  while (!idVerificado) {
    display.clearDisplay();
    display.setTextSize(1);
    display.setCursor(0, 0);
    display.println("Ingrese su cedula:");
    display.setCursor(0, 20);
    display.setTextSize(2);
    display.println(idTemp);
    display.setTextSize(1);
    display.setCursor(0, 50);
    display.println("[*]Borrar [#]Enviar");
    display.display();

    char key = keypad.getKey();

    if (key) {
      updateActivity();
      startTime = millis();

      if (key == '#') {
        // Validar longitud
        if (idTemp.length() >= MIN_ID_LENGTH && idTemp.length() <= MAX_ID_LENGTH) {
          Serial.println("🔍 Verificando ID: " + idTemp);

          // Mostrar mensaje de verificación con opción de cancelar
          display.clearDisplay();
          display.setTextSize(1);
          display.setCursor(0, 5);
          display.println("Verificando ID...");
          display.setCursor(0, 20);
          display.setTextSize(2);
          display.println(idTemp);
          display.setTextSize(1);
          display.setCursor(0, 40);
          display.println("Conectando...");
          display.setCursor(0, 55);
          display.println("[D] Cancelar");
          display.display();

          // VERIFICAR en Supabase con posibilidad de cancelar
          int resultado = verificarIDenSupabaseConCancelacion(idTemp);

          if (resultado == 1) {
```

```
      // ID verificado exitosamente
      Serial.println("✅ ID verificado y confirmado");
      idVerificado = true;
      return idTemp;
    } else if (resultado == 0) {
      // ID no encontrado en BD - Permitir reintentar
      mostrarPantallaNoRegistrado(idTemp);
      // Si retorna aquí (presionó [A]), limpiar y continuar
      idTemp = "";
      Serial.println("🔄 Limpiando campo para nuevo intento...");
      continue;  // Continuar el bucle para pedir ID de nuevo
    } else if (resultado == -1) {
      // Usuario canceló - Volver al menú principal
      Serial.println("🏠 Usuario canceló verificación");
      display.clearDisplay();
      display.setCursor(0, 20);
      display.println("Verificacion");
      display.setCursor(0, 35);
      display.println("cancelada");
      display.display();
      delay(1500);

      modeSelected = false;
      isOnline = false;
      idPacienteActual = "OFFLINE_USER";
      setup();  // Volver al inicio
      return "";
    }
  } else {
    Serial.println("❌ ID inválido (longitud incorrecta)");
    display.clearDisplay();
    display.setCursor(0, 10);
    display.println("ID invalido");
    display.setCursor(0, 25);
    display.print("Min: ");
    display.print(MIN_ID_LENGTH);
    display.print(" Max: ");
    display.println(MAX_ID_LENGTH);
    display.setCursor(0, 40);
    display.println("Intenta de nuevo");
    display.display();
    delay(2000);
    idTemp = "";
  }
} else if (key == '*') {
  if (idTemp.length() > 0) {
    idTemp.remove(idTemp.length() - 1);
  }
```

```
    } else if (key >= '0' && key <= '9') {
      if (idTemp.length() < MAX_ID_LENGTH) {
        idTemp += key;
      }
    }
  }

  // Timeout - Entrar en sleep en lugar de volver a llamar la función
  if (millis() - lastActivityTime > ID_INPUT_TIMEOUT) {
    Serial.println("⏰ Timeout en ingreso de cédula - Entrando en sleep");
    display.clearDisplay();
    display.setCursor(0, 20);
    display.println("Timeout");
    display.setCursor(0, 35);
    display.println("Entrando en sleep...");
    display.display();
    delay(2000);

    // Entrar en sleep
    particleSensor.shutDown();
    display.clearDisplay();
    display.display();

    esp_sleep_enable_ext0_wakeup((gpio_num_t)BUTTON_PIN, LOW);
    Serial.println("💤 Entrando en deep sleep...");
    delay(100);
    esp_deep_sleep_start();
  }

  delay(50);
 }

 return idTemp;
}

//=====================================================================
======
// CONFIGURAR WiFi CON AUTOCONNECT (FUNCIÓN MEJORADA)
//=====================================================================
======
bool configureWiFi() {
 Serial.println("╔═══════════════════════════════════════════════╗");
 Serial.println("║  INICIANDO WiFi CON AUTOCONNECT       ║");
 Serial.println("╚═══════════════════════════════════════════════╝");

 WiFi.mode(WIFI_STA);
```

```
//
═══════════════════════════════════════════════════════════════════════
// PASO 1: INTENTAR AUTOCONNECT CON CREDENCIALES GUARDADAS
//
═══════════════════════════════════════════════════════════════════════
display.clearDisplay();
display.setTextSize(1);
display.setCursor(0, 10);
display.println("WiFi AutoConnect");
display.setCursor(0, 25);
display.println("Buscando red");
display.setCursor(0, 40);
display.println("guardada...");
display.display();

Serial.println("📡 Intentando autoconnect...");

// WiFiManager guardará y cargará automáticamente las credenciales
WiFi.begin();  // Intenta conectar con credenciales guardadas

unsigned long startAttemptTime = millis();
int dots = 0;

// Esperar hasta WIFI_AUTOCONNECT_TIMEOUT (15 seg)
while (WiFi.status() != WL_CONNECTED &&
       millis() - startAttemptTime < WIFI_AUTOCONNECT_TIMEOUT) {

  delay(500);
  Serial.print(".");

  // Animación en pantalla
  if (++dots > 3) dots = 0;
  display.clearDisplay();
  display.setCursor(0, 10);
  display.println("Conectando WiFi");
  display.setCursor(0, 25);
  display.print("Autoconnect");
  for(int i = 0; i < dots; i++) {
    display.print(".");
  }
  display.setCursor(0, 45);
  display.print("Tiempo: ");
  display.print((millis() - startAttemptTime) / 1000);
  display.println("s");
  display.display();
}
```

```
  Serial.println();

  //
═══════════════════════════════════════════════════════════════════════
═══════════════════
  // ¿AUTOCONNECT EXITOSO?
  //
═══════════════════════════════════════════════════════════════════════
═══════════════════
  if (WiFi.status() == WL_CONNECTED) {
    Serial.println("✅ AUTOCONNECT EXITOSO!");
    Serial.print("📶 Conectado a: ");
    Serial.println(WiFi.SSID());
    Serial.print("📶 IP: ");
    Serial.println(WiFi.localIP());
    Serial.print("📶 Señal: ");
    Serial.print(WiFi.RSSI());
    Serial.println(" dBm");

    // ⬅️ Encender LED azul al conectar WiFi
    digitalWrite(LED_PIN, HIGH);

    display.clearDisplay();
    display.setCursor(0, 5);
    display.println("WiFi Conectado!");
    display.setCursor(0, 20);
    display.println("Red: " + WiFi.SSID());
    display.setCursor(0, 35);
    display.print("IP: ");
    display.println(WiFi.localIP());
    display.display();
    delay(2500);

    return true;
  }

  //
═══════════════════════════════════════════════════════════════════════
═══════════════════
  // PASO 2: AUTOCONNECT FALLÓ → INICIAR PORTAL CAUTIVO
  //
═══════════════════════════════════════════════════════════════════════
═══════════════════
  Serial.println("⚠️ Autoconnect falló - Iniciando portal cautivo");

  display.clearDisplay();
  display.setCursor(0, 5);
  display.println("No hay red guardada");
```

```
display.setCursor(0, 20);
display.println("Iniciando portal");
display.setCursor(0, 35);
display.println("cautivo...");
display.display();
delay(2000);

WiFiManager wifiManager;
wifiManager.setConfigPortalTimeout(WIFI_CONFIG_TIMEOUT / 1000);  // Convertir a segundos

display.clearDisplay();
display.setTextSize(1);
display.setCursor(0, 0);
display.println("Portal WiFi activo");
display.setCursor(0, 15);
display.println("Conectate a:");
display.setTextSize(2);
display.setCursor(0, 30);
display.println("BpmSp02");
display.setTextSize(1);
display.setCursor(0, 50);
display.println("Config en 192.168.4.1");
display.display();

Serial.println("╔═══════════════════════════════════════╗");
Serial.println("║     PORTAL CAUTIVO ACTIVO             ║");
Serial.println("╚═══════════════════════════════════════╝");
Serial.println("📱 Conéctate al WiFi: BpmSp02");
Serial.println("🌐 Abre: 192.168.4.1");
Serial.println("⏱️ Timeout: 3 minutos");

// WiFiManager intentará conectar o abrirá portal
if (!wifiManager.autoConnect("BpmSp02")) {
  Serial.println("❌ Error: No se configuró WiFi en el tiempo límite");

  display.clearDisplay();
  display.setCursor(0, 15);
  display.println("Timeout WiFi");
  display.setCursor(0, 30);
  display.println("Reiniciando...");
  display.display();
  delay(3000);

  ESP.restart();
  return false;
}
```

```
  //
═══════════════════════════════════════════════════════════════════════
  // CONEXIÓN EXITOSA VÍA PORTAL
  //
═══════════════════════════════════════════════════════════════════════
  Serial.println("✅ WiFi configurado exitosamente!");
  Serial.print("📶 Conectado a: ");
  Serial.println(WiFi.SSID());
  Serial.print("📶 IP: ");
  Serial.println(WiFi.localIP());
  Serial.print("📶 Señal: ");
  Serial.print(WiFi.RSSI());
  Serial.println(" dBm");

  // ⬅️ Encender LED azul al conectar WiFi
  digitalWrite(LED_PIN, HIGH);

  display.clearDisplay();
  display.setCursor(0, 5);
  display.println("WiFi Conectado!");
  display.setCursor(0, 20);
  display.println("Red: " + WiFi.SSID());
  display.setCursor(0, 35);
  display.print("IP: ");
  display.println(WiFi.localIP());
  display.display();
  delay(2500);

  return true;
}

//=========================================================================
======
// ENVIAR DATOS A SUPABASE
//=========================================================================
======
void sendToSupabase(float bpm, float spo2) {
  // Verificar conexión WiFi antes de enviar
  checkAndMaintainWiFi();

  if (WiFi.status() != WL_CONNECTED) {
    Serial.println("❌ No se puede enviar: WiFi desconectado");
    return;
  }

  // ⬅️ VALIDACIÓN ENFOCADA EN BPM
```

```
// BPM válido: 40-180 (rango médico realista en reposo y ejercicio moderado)
// SpO2 válido: 70-100 (rango amplio, el sensor es confiable aquí)
if (bpm < 40 || bpm > 180) {
  Serial.print("❌ BPM fuera de rango válido: ");
  Serial.println(bpm);
  return;
}

if (spo2 < 70 || spo2 > 100) {
  Serial.print("❌ SpO2 fuera de rango válido: ");
  Serial.println(spo2);
  return;
}

// Verificar que tengamos suficientes lecturas válidas
if (validReadings < MIN_VALID_READINGS) {
  Serial.println("❌ No hay suficientes lecturas válidas aún");
  return;
}

HTTPClient http;

Serial.println("╔═══════════════════════════════════════════╗");
Serial.println("║     ENVIANDO DATOS A SUPABASE             ║");
Serial.println("╚═══════════════════════════════════════════╝");
Serial.print("📍 URL: ");
Serial.println(SUPABASE_URL);
Serial.print("👤 Cédula: ");
Serial.println(idPacienteActual);
Serial.print("💓 BPM: ");
Serial.println(bpm, 0);
Serial.print("🫁 SpO2: ");
Serial.println(spo2, 1);

http.begin(SUPABASE_URL);

http.addHeader("Content-Type", "application/json");
http.addHeader("apikey", SUPABASE_KEY);
http.addHeader("Authorization", String("Bearer ") + SUPABASE_KEY);
http.addHeader("Prefer", "return=minimal");

// ⬅️ CORREGIDO: Usar nombres de columnas correctos
StaticJsonDocument<200> doc;
doc["cedula_paciente"] = idPacienteActual;  // ← Nombre correcto
doc["ritmo_cardiaco"] = bpm;                // ← Nombre correcto
doc["spo2"] = spo2;                         // ← Nombre correcto

String jsonData;
```

```
  serializeJson(doc, jsonData);

  Serial.println("📦 JSON a enviar:");
  Serial.println(jsonData);

  int httpResponseCode = http.POST(jsonData);

  Serial.print("📥 Código de respuesta: ");
  Serial.println(httpResponseCode);

  if (httpResponseCode > 0) {
    String response = http.getString();
    Serial.println("📄 Respuesta completa:");
    Serial.println(response);

    if (httpResponseCode == 201) {
      Serial.println("✅ Datos enviados exitosamente!");
    } else if (httpResponseCode == 400) {
      Serial.println("❌ ERROR 400: Bad Request");
      Serial.println("Posibles causas:");
      Serial.println("- Formato JSON incorrecto");
      Serial.println("- Nombres de columnas incorrectos");
      Serial.println("- Tipos de datos no coinciden");
      Serial.println("- Falta campo requerido en BD");
    } else if (httpResponseCode == 401) {
      Serial.println("❌ ERROR 401: No autorizado - Revisa API Key");
    } else if (httpResponseCode == 403) {
      Serial.println("❌ ERROR 403: Prohibido - Revisa políticas RLS");
    } else {
      Serial.print("⚠️ Código inesperado: ");
      Serial.println(httpResponseCode);
    }
  } else {
    Serial.print("❌ Error en la petición HTTP: ");
    Serial.println(httpResponseCode);
    Serial.println("Posibles causas:");
    Serial.println("- Conexión WiFi inestable");
    Serial.println("- URL incorrecta");
    Serial.println("- Timeout de red");
  }

  http.end();
  Serial.println("═══════════════════════════════════════════════════════\n");
}

//=========================================================================
======
// RESETEAR VARIABLES DE MEDICIÓN
```

```
//=============================================================================
======
void resetVariables() {
  Serial.println("🔄 Reseteando variables");

  for (byte j = 0; j < RATE_SIZE; j++) {
    rates[j] = 0;
  }

  rateSpot = 0;
  beatAvg = 0;
  lastBeat = 0;

  avered = 0;
  aveir = 0;
  sumirrms = 0;
  sumredrms = 0;
  i = 0;
  ESpO2 = 0;
  smoothedR = 1.0;

  // ⬅️ NUEVO: Resetear contador de lecturas válidas
  validReadings = 0;
  sensorWarmedUp = false;

  display.clearDisplay();
  display.setCursor(0, 20);
  display.println("Variables");
  display.setCursor(0, 35);
  display.println("reseteadas");
  display.display();
  delay(1000);
}

//=============================================================================
======
// ENTRAR EN MODO BAJO CONSUMO (CON MODO SELECCIONADO)
//=============================================================================
======
void enterLowPowerMode() {
  lowPowerMode = true;

  Serial.println("╔═══════════════════════════════════════╗");
  Serial.println("║  ENTRANDO EN MODO SLEEP       ║");
  Serial.println("╚═══════════════════════════════════════╝");

  display.clearDisplay();
  display.setTextSize(1);
```

```
  display.setCursor(0, 15);
  display.println("Modo bajo consumo");
  display.setCursor(0, 30);
  display.println("Presiona boton");
  display.setCursor(0, 45);
  display.println("para despertar");
  display.display();
  delay(LOW_POWER_INFO_DELAY);

  particleSensor.shutDown();
  display.clearDisplay();
  display.display();

  esp_sleep_enable_ext0_wakeup((gpio_num_t)BUTTON_PIN, LOW);
  Serial.println("💤 Entrando en deep sleep...");
  delay(100);
  esp_deep_sleep_start();
}

//=====================================================================
======
// ENTRAR EN MODO BAJO CONSUMO (SIN MODO - TIMEOUT SETUP)
//=====================================================================
======
void enterLowPowerModeNoMode() {
  Serial.println("╔═══════════════════════════════════════════╗");
  Serial.println("║  TIMEOUT - ENTRANDO EN SLEEP ║");
  Serial.println("╚═══════════════════════════════════════════╝");

  display.clearDisplay();
  display.setTextSize(1);
  display.setCursor(0, 20);
  display.println("Sin actividad");
  display.setCursor(0, 35);
  display.println("Entrando en sleep");
  display.display();
  delay(2000);

  particleSensor.shutDown();
  display.clearDisplay();
  display.display();

  esp_sleep_enable_ext0_wakeup((gpio_num_t)BUTTON_PIN, LOW);
  Serial.println("💤 Entrando en deep sleep...");
  delay(100);
  esp_deep_sleep_start();
}
```

```
//=============================================================================
======
// MOSTRAR MEDICIONES EN PANTALLA
//=============================================================================
======
void displayMeasurements(float bpm, float spo2) {
  display.clearDisplay();
  display.setTextSize(1);

  // Estado WiFi en esquina superior (solo si online)
  if (isOnline) {
    if (WiFi.status() == WL_CONNECTED) {
      int rssi = WiFi.RSSI();
      display.setCursor(110, 0);
      if (rssi > -60) {
        display.println("|||");  // Señal excelente
      } else if (rssi > -70) {
        display.println("|| ");  // Señal buena
      } else if (rssi > -80) {
        display.println("|  ");  // Señal regular
      } else {
        display.println("!  ");  // Señal mala
      }
    } else {
      display.setCursor(110, 0);
      display.println("X");  // Desconectado
    }
  }

  // BPM
  display.setCursor(0, 0);
  display.println("BPM:");
  display.setTextSize(3);
  display.setCursor(0, 12);

  // ⬅️ MEJORADO: Mostrar -- si BPM no es confiable
  if (bpm >= 40 && bpm <= 180 && validReadings >= MIN_VALID_READINGS) {
    display.println((int)bpm);
  } else {
    display.println("--");
  }

  // SpO2
  display.setTextSize(1);
  display.setCursor(0, 38);
  display.println("SpO2:");
  display.setTextSize(2);
  display.setCursor(0, 48);
```

```
  if (spo2 >= 70 && spo2 <= 100 && validReadings >= MIN_VALID_READINGS) {
    display.print((int)spo2);
    display.println("%");
  } else {
    display.println("--%");
  }

  // ⬅️ NUEVO: Agregar indicación [D] Menú en la parte inferior
  display.setTextSize(1);
  display.setCursor(75, 56);
  display.println("[D]Menu");

  display.display();

  Serial.print("❤️ BPM: ");
  Serial.print(bpm, 0);
  Serial.print(" | 🫁 SpO2: ");
  Serial.print(spo2, 1);
  Serial.println("%");
}

//================================================================================
// MOSTRAR SELECCIÓN DE MODO
//================================================================================
void displayModeSelection() {
  display.clearDisplay();
  display.setTextSize(1);
  display.setCursor(0, 0);
  display.println("Seleccione modo:");
  display.setCursor(0, 20);
  display.println("[1] Online");
  display.setCursor(0, 30);
  display.println("    (Supabase Cloud)");
  display.setCursor(0, 45);
  display.println("[0] Offline");
  display.setCursor(0, 55);
  display.println("    (Uso en casa)");
  display.display();
}

//================================================================================
// PROCESAR TECLA DURANTE MEDICIÓN
//================================================================================
```

```
void processKeyDuringMeasurement(char key) {
  updateActivity();

  if (key == '#') {
    // Sleep
    Serial.println("⌨️  Tecla [#] - Entrando en sleep");
    enterLowPowerMode();
  } else if (key == '*') {
    // Reset
    Serial.println("⌨️  Tecla [*] - Reset variables");
    resetVariables();
  } else if (key == 'D') {
    // Menú de configuración
    Serial.println("⌨️  Tecla [D] - Menú");
    displayConfigMenu();
  }
}

//================================================================================
======
// MOSTRAR MENÚ DE CONFIGURACIÓN
//================================================================================
======
void displayConfigMenu() {
  bool inMenu = true;
  unsigned long menuStartTime = millis();

  while (inMenu) {
    display.clearDisplay();
    display.setTextSize(1);
    display.setCursor(0, 0);
    display.println("=== MENU ===");
    display.setCursor(0, 12);
    display.println("[1] Ver info");
    display.setCursor(0, 22);
    display.println("[2] Reset variables");
    display.setCursor(0, 32);
    display.println("[3] Cambiar modo");

    // Solo mostrar Reset WiFi si está en modo Online
    if (isOnline) {
      display.setCursor(0, 42);
      display.println("[4] Reset WiFi");
    }

    display.setCursor(0, 52);
    display.println("[#]Sleep [D]Salir");
    display.display();
```

```
  char key = keypad.getKey();

  if (key) {
    updateActivity();
    menuStartTime = millis();

    if (key == '1') {
      //
═══════════════════════════════════════════════════════════════════════
      // OPCIÓN 1: VER INFORMACIÓN
      //
═══════════════════════════════════════════════════════════════════════
      bool viewingInfo = true;

      while (viewingInfo) {
        display.clearDisplay();
        display.setCursor(0, 0);
        display.println("=== INFO ===");
        display.setCursor(0, 12);
        display.print("Modo: ");
        display.println(isOnline ? "ONLINE" : "OFFLINE");
        display.setCursor(0, 22);
        display.print("ID: ");
        display.println(idPacienteActual);

        if (isOnline && WiFi.status() == WL_CONNECTED) {
          display.setCursor(0, 32);
          display.println("WiFi: " + WiFi.SSID());
          display.setCursor(0, 42);
          display.print("Senal: ");
          display.print(WiFi.RSSI());
          display.println("dBm");
        } else if (isOnline) {
          display.setCursor(0, 32);
          display.println("WiFi: Desconectado");
        }

        display.setCursor(0, 56);
        display.println("[D] Volver");
        display.display();

        char infoKey = keypad.getKey();
        if (infoKey == 'D') {
          viewingInfo = false;
          updateActivity();
        }
```

```
      if (checkMenuTimeout(MENU_TIMEOUT)) {
        return;
      }

      delay(50);
    }

  } else if (key == '2') {
    //
═══════════════════════════════════════════════════════════════
    // OPCIÓN 2: RESET VARIABLES
    //
═══════════════════════════════════════════════════════════════
    resetVariables();
    delay(1000);

  } else if (key == '3') {
    //
═══════════════════════════════════════════════════════════════
    // OPCIÓN 3: CAMBIAR MODO (NUEVA FUNCIONALIDAD)
    //
═══════════════════════════════════════════════════════════════
    display.clearDisplay();
    display.setCursor(0, 5);
    display.println("Cambiar modo?");
    display.setCursor(0, 20);
    display.print("Actual: ");
    display.println(isOnline ? "ONLINE" : "OFFLINE");
    display.setCursor(0, 35);
    display.println("[1] Cambiar a " + String(!isOnline ? "ONLINE" : "OFFLINE"));
    display.setCursor(0, 50);
    display.println("[D] Cancelar");
    display.display();

    bool waitingModeChange = true;
    while (waitingModeChange) {
      char modeKey = keypad.getKey();

      if (modeKey == '1') {
        // Confirmar cambio de modo
        display.clearDisplay();
        display.setCursor(0, 15);
        display.println("Cambiando modo...");
        display.setCursor(0, 30);
        display.println("Reiniciando...");
        display.display();

        Serial.println("🔄 Cambiando de modo - Reiniciando sistema");
```

```
      delay(2000);
      ESP.restart();  // Reiniciar para cambiar de modo limpiamente

    } else if (modeKey == 'D') {
      waitingModeChange = false;
      updateActivity();
    }

    if (checkMenuTimeout(MENU_TIMEOUT)) {
      return;
    }

    delay(50);
  }

} else if (key == '4' && isOnline) {
  //
═══════════════════════════════════════════════════════════════
  // OPCIÓN 4: RESET WiFi (solo Online)
  //
═══════════════════════════════════════════════════════════════
  display.clearDisplay();
  display.setCursor(0, 10);
  display.println("Confirmar reset?");
  display.setCursor(0, 25);
  display.println("[1] Si, resetear");
  display.setCursor(0, 40);
  display.println("[D] Cancelar");
  display.display();

  bool waitingConfirm = true;
  unsigned long confirmStart = millis();

  while (waitingConfirm) {
    char confirmKey = keypad.getKey();

    if (confirmKey == '1') {
      display.clearDisplay();
      display.setCursor(0, 20);
      display.println("Borrando config");
      display.setCursor(0, 35);
      display.println("WiFi...");
      display.display();

      WiFiManager wifiManager;
      wifiManager.resetSettings();

      Serial.println("🔄 WiFi reseteado - Reiniciando...");
```

```
          delay(2000);
          ESP.restart();

        } else if (confirmKey == 'D') {
          waitingConfirm = false;
          updateActivity();
        }

        if (millis() - confirmStart > 10000) {
          waitingConfirm = false;
        }

        delay(50);
      }

    } else if (key == '#') {
      //
═══════════════════════════════════════════════════════════════════════
      // OPCIÓN #: SLEEP
      //
═══════════════════════════════════════════════════════════════════════
      inMenu = false;
      enterLowPowerMode();

    } else if (key == 'D') {
      //
═══════════════════════════════════════════════════════════════════════
      // OPCIÓN D: SALIR DEL MENÚ
      //
═══════════════════════════════════════════════════════════════════════
      inMenu = false;
      Serial.println("⌨️ Saliendo del menú");
    }
  }

  if (checkMenuTimeout(MENU_TIMEOUT)) {
    return;
  }

  delay(50);
 }
}

//=====================================================================
======
// SETUP
//=====================================================================
======
```

```
void setup() {
  Serial.begin(115200);
  Serial.println("\n\n╔═══════════════════════════════════════════════════════════╗");
  Serial.println("║     BpmSp02 v1.9 - WiFi AutoConnect     ║");
  Serial.println("╚═══════════════════════════════════════════════════════════╝\n");

  pinMode(BUTTON_PIN, INPUT_PULLUP);
  pinMode(LED_PIN, OUTPUT);
  digitalWrite(LED_PIN, LOW);  // ⬅️ Inicia apagado - solo enciende con WiFi conectado

  Wire.begin();
  display.begin(SCREEN_ADDR);
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(WHITE);

  // Pantalla de bienvenida
  display.clearDisplay();
  display.setTextSize(2);
  display.setCursor(10, 10);
  display.println("BpmSp02");
  display.setTextSize(1);
  display.setCursor(20, 35);
  display.println("Version 1.9");
  display.setCursor(10, 50);
  display.println("Bienvenido!");
  display.display();
  delay(WELCOME_DELAY);

  // Verificar si hay despertar desde deep sleep
  esp_sleep_wakeup_cause_t wakeup_reason = esp_sleep_get_wakeup_cause();

  if (wakeup_reason == ESP_SLEEP_WAKEUP_EXT0) {
    Serial.println("═══════════════════════════════════════════════════════");
    Serial.println("  DESPERTANDO DESDE DEEP SLEEP");
    Serial.println("═══════════════════════════════════════════════════════");

    // Si estaba en Online, volver a pedir cédula
    waitingForWakeDecision = true;

    display.clearDisplay();
    display.setTextSize(1);
    display.setCursor(0, 10);
    display.println("Sistema despertado");
    display.setCursor(0, 30);
    display.println("[1] Continuar Online");
    display.setCursor(0, 45);
    display.println("[0] Continuar Offline");
```

```
display.display();

// ⬅️ IMPORTANTE: Usar timer independiente para wake decision
unsigned long wakeStartTime = millis();

while (waitingForWakeDecision) {
  char key = keypad.getKey();

  if (key) {
    if (key == '1') {
      isOnline = true;
      modeSelected = true;
      waitingForWakeDecision = false;

      // Reconectar WiFi
      display.clearDisplay();
      display.setCursor(0, 20);
      display.println("Reconectando WiFi...");
      display.display();

      if (configureWiFi()) {
        // Pedir cédula de nuevo (SEGURIDAD)
        idPacienteActual = solicitarIDPaciente();

        display.clearDisplay();
        display.setCursor(0, 20);
        display.println("Cedula verificada:");
        display.setCursor(0, 35);
        display.println(idPacienteActual);
        display.display();
        delay(2000);
      } else {
        ESP.restart();
      }

    } else if (key == '0') {
      isOnline = false;
      modeSelected = true;
      waitingForWakeDecision = false;
      idPacienteActual = "OFFLINE_USER";

      display.clearDisplay();
      display.setCursor(0, 20);
      display.println("Modo Offline");
      display.setCursor(0, 35);
      display.println("Sin conexion");
      display.display();
      delay(1500);
```

```
    }
  }

  // ⬅️ Timeout de 30 seg - Volver a SLEEP (no continuar en Offline)
  if (millis() - wakeStartTime > WAKE_DECISION_TIMEOUT) {
    display.clearDisplay();
    display.setCursor(0, 15);
    display.println("Timeout!");
    display.setCursor(0, 35);
    display.println("Volviendo a sleep...");
    display.display();
    delay(2000);

    Serial.println("⏰ Timeout - Volviendo a sleep");

    // Apagar pantalla
    display.clearDisplay();
    display.display();

    // Configurar wake up y entrar en sleep
    esp_sleep_enable_ext0_wakeup((gpio_num_t)BUTTON_PIN, LOW);
    Serial.println("💤 Entrando en deep sleep desde wake timeout...");
    delay(100);
    esp_deep_sleep_start();
  }

  delay(50);
}

// ⬅️ CRÍTICO: Resetear lastActivityTime después de seleccionar modo
// Esto evita que entre inmediatamente en sleep por el LOW_POWER_TIMEOUT
lastActivityTime = millis();
lastDetectedTime = millis();
Serial.println("✅ Timers de actividad reseteados después de wake");


// Reconfigurar sensor después de despertar
if (!particleSensor.begin(Wire, I2C_SPEED_FAST)) {
  Serial.println("❌ Error: No se detectó MAX30102");
  display.clearDisplay();
  display.setCursor(0, 20);
  display.println("Error sensor");
  display.display();
  while (1);
}

particleSensor.wakeUp();
delay(100);
```

```
  particleSensor.setup(0x5A, 4, 2, 400, 411, 16384);
  particleSensor.enableDIETEMPRDY();

  for (byte j = 0; j < RATE_SIZE; j++) {
    rates[j] = 0;
  }

  updateActivity();

  Serial.println("✅ Sistema listo después de wake up");
  return;
}

// INICIO NORMAL (no es wake up)
Serial.println("═══════════════════════════════════════════════════");
Serial.println("  INICIO NORMAL DEL SISTEMA");
Serial.println("═══════════════════════════════════════════════════");

// Configurar MAX30102
Serial.println("Inicializando MAX30102...");
if (!particleSensor.begin(Wire, I2C_SPEED_FAST)) {
  Serial.println("❌ Error: No se detectó MAX30102");
  display.clearDisplay();
  display.setCursor(0, 20);
  display.println("Error: No hay");
  display.setCursor(0, 35);
  display.println("sensor MAX30102");
  display.display();
  while (1);
}

Serial.println("✅ MAX30102 detectado");
particleSensor.shutDown();

//
═══════════════════════════════════════════════════════════════════
═══════════════════
// PASO 1: SELECCIONAR MODO PRIMERO (NUEVO FLUJO)
//
═══════════════════════════════════════════════════════════════════
═══════════════════
displayModeSelection();
updateActivity();

Serial.println("=== PASO 1: Seleccionar modo ===");
Serial.println("Esperando modo (1=Online, 0=Offline)");

while (!modeSelected) {
```

```
  char key = keypad.getKey();

  if (key) {
    updateActivity();

    if (key == '1') {
      isOnline = true;
      modeSelected = true;
      Serial.println("✓ Modo ONLINE seleccionado");
    } else if (key == '0') {
      isOnline = false;
      modeSelected = true;
      Serial.println("✓ Modo OFFLINE seleccionado");
    }
  }

  // Timeout en selección de modo - Entrar en sleep en lugar de reiniciar
  if (millis() - lastActivityTime > MODE_SELECT_TIMEOUT) {
    Serial.println("⏰ Timeout en selección de modo - Entrando en sleep");
    display.clearDisplay();
    display.setCursor(0, 20);
    display.println("Timeout");
    display.setCursor(0, 35);
    display.println("Entrando en sleep...");
    display.display();
    delay(2000);

    // Entrar en sleep sin haber seleccionado modo
    particleSensor.shutDown();
    display.clearDisplay();
    display.display();

    esp_sleep_enable_ext0_wakeup((gpio_num_t)BUTTON_PIN, LOW);
    Serial.println("💤 Entrando en deep sleep...");
    delay(100);
    esp_deep_sleep_start();
  }

  delay(50);
}

// Mostrar modo seleccionado
display.clearDisplay();
display.setCursor(0, 0);
display.println("Modo seleccionado:");
display.setCursor(0, 20);
display.setTextSize(2);
if (isOnline) {
```

```
  display.println("ONLINE");
  display.setTextSize(1);
  display.setCursor(0, 40);
  display.println("Supabase Cloud");
 } else {
  display.println("OFFLINE");
  display.setTextSize(1);
  display.setCursor(0, 40);
  display.println("Uso en casa");
 }
 display.display();
 delay(MODE_DISPLAY_DELAY);

 //
═══════════════════════════════════════════════════════════════════════
═══════════════════
 // PASO 2: SI ES ONLINE → CONFIGURAR WIFI CON AUTOCONNECT
 //
═══════════════════════════════════════════════════════════════════════
═══════════════════
 if (isOnline) {
  Serial.println("=== PASO 2: Configurar WiFi (modo Online) ===");

  // Configurar WiFi con autoconnect ANTES de pedir cédula
  bool wifiConfigured = configureWiFi();
  if (!wifiConfigured) {
   Serial.println("WiFi no configurado - Reiniciando");
   delay(2000);
   ESP.restart();
   return;
  }

  //
═══════════════════════════════════════════════════════════════════════
═══════════════════
  // PASO 3: AHORA SÍ PEDIR CÉDULA (YA HAY WIFI PARA VERIFICAR)
  //
═══════════════════════════════════════════════════════════════════════
═══════════════════
  Serial.println("=== PASO 3: Solicitar y verificar cédula ===");

  // Pedir cédula (incluye verificación automática en BD)
  idPacienteActual = solicitarIDPaciente();
  Serial.println("✓ Cédula verificada: " + idPacienteActual);

  display.clearDisplay();
  display.setCursor(0, 20);
  display.println("Cedula verificada:");
```

```
  display.setCursor(0, 35);
  display.setTextSize(1);
  display.println(idPacienteActual);
  display.display();
  delay(2000);

 } else {
  // Modo Offline: NO pide cédula ni WiFi
  Serial.println("=== PASO 2: Modo Offline - Sin cédula ni WiFi ===");
  idPacienteActual = "OFFLINE_USER";

  // ⬅️ Asegurar que LED esté apagado en modo Offline
  digitalWrite(LED_PIN, LOW);

  display.clearDisplay();
  display.setCursor(0, 15);
  display.println("Modo OFFLINE activo");
  display.setCursor(0, 30);
  display.println("No requiere cedula");
  display.setCursor(0, 45);
  display.println("Uso personal");
  display.display();
  delay(2000);
 }

 //
═══════════════════════════════════════════════════════════════════════
═══════════════════
 // PASO 4: CONFIGURAR SENSOR
 //
═══════════════════════════════════════════════════════════════════════
═══════════════════
 Serial.println("=== Configurando sensor ===");
 particleSensor.wakeUp();
 delay(100);
 particleSensor.setup(0x5A, 4, 2, 400, 411, 16384);
 particleSensor.enableDIETEMPRDY();

 for (byte j = 0; j < RATE_SIZE; j++) {
  rates[j] = 0;
 }

 display.clearDisplay();
 display.setTextSize(1);
 display.setCursor(0, 20);
 display.println("Sensor listo!");
 display.setCursor(0, 35);
 display.println("Coloca tu dedo");
```

```
  display.display();
  delay(1500);

  updateActivity();

  Serial.println("╔═══════════════════════════════════════╗");
  Serial.println("║     SISTEMA LISTO          ║");
  Serial.println("╚═══════════════════════════════════════╝");
  Serial.println("Modo: " + String(isOnline ? "ONLINE" : "OFFLINE"));
  Serial.println("ID: " + idPacienteActual);
  if (isOnline) {
    Serial.println("WiFi: " + WiFi.SSID());
    Serial.println("AutoConnect: ACTIVO ✅");
  }
}

//=====================================================================
======
// LOOP PRINCIPAL
//=====================================================================
======
void loop() {
  if (!modeSelected) {
    delay(100);
    return;
  }

  char key = keypad.getKey();
  if (key) {
    Serial.print("Tecla: ");
    Serial.println(key);
    processKeyDuringMeasurement(key);
  }

  long ir = particleSensor.getIR();
  long red = particleSensor.getRed();

  if (ir < FINGER_ON) {
    //
═══════════════════════════════════════════════════════════════════
═══════════
    // NO HAY DEDO DETECTADO - Es seguro revisar WiFi aquí
    //
═══════════════════════════════════════════════════════════════════
═══════════
    if (isOnline) {
      checkAndMaintainWiFi();  // Solo se ejecuta cada 60 seg
    }
```

```
  if (millis() - lastPrintTime > NO_FINGER_PRINT_INTERVAL) {
    display.clearDisplay();
    display.setTextSize(1);  // ← Tamaño 1 para que quepa el mensaje
    display.setCursor(5, 15);
    display.println("No hay dedo");
    display.setCursor(5, 30);
    display.println("detectado");
    display.setCursor(0, 50);
    display.println("[#]Sleep [D]Menu");
    display.display();
    lastPrintTime = millis();
  }

  if (isFingerDetected && millis() - lastDetectedTime > RESET_VARS_TIMEOUT) {
    resetVariables();
    isFingerDetected = false;
  }

  if (!lowPowerMode && millis() - lastDetectedTime > LOW_POWER_TIMEOUT) {
    enterLowPowerMode();
  }

  return;
 }

 //
═══════════════════════════════════════════════════════════════════════
═══════════
  // HAY DEDO DETECTADO - PRIORIZAR MEDICIONES, NO REVISAR WiFi
  //
═══════════════════════════════════════════════════════════════════════
═══════════
  isFingerDetected = true;
  lastDetectedTime = millis();
  updateActivity();

  // ⬅️ NUEVO: Verificar calidad de la señal
  // Si IR o RED son demasiado bajos o altos, la lectura no será confiable
  if (ir < 50000 || ir > 200000 || red < 10000 || red > 200000) {
    if (millis() - lastPrintTime > 2000) {
      display.clearDisplay();
      display.setCursor(0, 10);
      display.println("Ajusta el dedo");
      display.setCursor(0, 25);
      display.println("Senal debil o");
      display.setCursor(0, 40);
      display.println("demasiado fuerte");
```

```
    display.display();

    Serial.print("⚠️ Señal fuera de rango - IR: ");
    Serial.print(ir);
    Serial.print(" RED: ");
    Serial.println(red);

    lastPrintTime = millis();
  }

  // Resetear variables si la señal es mala
  if (validReadings > 0) {
    validReadings = 0;
    Serial.println("🔄 Reseteando por señal mala");
  }

  return;
}

// Revisar y reportar señal WiFi cada 10 segundos (solo log, sin reconexión)
if (isOnline && (millis() - lastSignalCheckTime > 10000)) {
  if (WiFi.status() == WL_CONNECTED) {
    int rssi = WiFi.RSSI();
    Serial.print("📶 Señal WiFi actual: ");
    Serial.print(rssi);
    Serial.print(" dBm - ");
    if (rssi > -60) {
      Serial.println("Excelente ✅");
    } else if (rssi > -70) {
      Serial.println("Buena ✅");
    } else if (rssi > -80) {
      Serial.println("Regular ⚠️");
    } else {
      Serial.println("Mala ❌ - Acércate al router");
    }
  } else {
    Serial.println("⚠️ WiFi desconectado - Se reconectará cuando no haya medición");
  }
  lastSignalCheckTime = millis();
}

if (checkForBeat(ir)) {
  long delta = millis() - lastBeat;
  lastBeat = millis();
  float beatsPerMinute = 60 / (delta / 1000.0);

  // ⬅️ MEJORADO: Validación más estricta de BPM
  // Rango realista: 40-180 BPM
```

```
  // Evitar lecturas erróneas por movimiento o mala colocación
  if (beatsPerMinute >= 40 && beatsPerMinute <= 180) {
    rates[rateSpot++] = (byte)beatsPerMinute;
    rateSpot %= RATE_SIZE;

    beatAvg = 0;
    byte validRates = 0;

    // Calcular promedio solo con valores válidos
    for (byte j = 0; j < RATE_SIZE; j++) {
      if (rates[j] >= 40 && rates[j] <= 180) {
        beatAvg += rates[j];
        validRates++;
      }
    }

    if (validRates > 0) {
      beatAvg /= validRates;
    }

    Serial.print("💓 Latido detectado: ");
    Serial.print(beatsPerMinute, 0);
    Serial.print(" BPM | Promedio: ");
    Serial.print(beatAvg, 0);
    Serial.println(" BPM");
  } else {
    Serial.print("⚠️ BPM descartado (fuera de rango): ");
    Serial.println(beatsPerMinute, 0);
  }
}

avered = avered * frate + red * (1.0 - frate);
aveir = aveir * frate + ir * (1.0 - frate);
sumredrms += pow((red - avered), 2);
sumirrms += pow((ir - aveir), 2);

if ((i % Num) == 0) {
  double rawR = (sqrt(sumredrms) / avered) / (sqrt(sumirrms) / aveir);

  // ⬅️ VALIDACIÓN: Verificar que la razón R sea razonable
  // R típicamente está entre 0.4 y 2.0 para lecturas válidas
  if (rawR < 0.4 || rawR > 2.0) {
    Serial.print("⚠️ Razón R fuera de rango: ");
    Serial.println(rawR);
    sumredrms = 0.0;
    sumirrms = 0.0;
    i = 0;
    return; // Descartar esta lectura
```

```
}

smoothedR = alpha * smoothedR + (1 - alpha) * rawR;

// ⬅️ MEJORADO: Cálculo de SpO2 con validación
float SpO2 = -25 * smoothedR + 110;

// Validar rango realista de SpO2
if (SpO2 < 70 || SpO2 > 100) {
  Serial.print("⚠️ SpO2 calculado fuera de rango: ");
  Serial.print(SpO2);
  Serial.print("% (R=");
  Serial.print(smoothedR);
  Serial.println(")");

  // No actualizar ESpO2 con valores incorrectos
  sumredrms = 0.0;
  sumirrms = 0.0;
  i = 0;
  return;
}

// Incrementar contador de lecturas válidas
validReadings++;

// Solo actualizar ESpO2 después de varias lecturas válidas
if (validReadings >= MIN_VALID_READINGS) {
  ESpO2 = FSpO2 * ESpO2 + (1.0 - FSpO2) * SpO2;

  // Asegurar que ESpO2 esté en rango válido
  ESpO2 = constrain(ESpO2, 70, 100);

  displayMeasurements(beatAvg, ESpO2);

  if (isOnline && (millis() - lastSendTime > DATA_SEND_INTERVAL)) {
    // Intentar enviar solo si WiFi está conectado
    if (WiFi.status() == WL_CONNECTED) {
      sendToSupabase(beatAvg, ESpO2);
      lastSendTime = millis();

      // ⬅️ Parpadeo LED al enviar datos (más visible)
      digitalWrite(LED_PIN, LOW);
      delay(200);
      digitalWrite(LED_PIN, HIGH);
    } else {
      Serial.println("⚠️ Datos no enviados - WiFi desconectado");
      // ⬅️ LED apagado si no hay WiFi
      digitalWrite(LED_PIN, LOW);
```

```
      }
    }
  } else {
    // Mostrar mensaje de calibración
    display.clearDisplay();
    display.setTextSize(1);  // ← Todo en tamaño 1
    display.setCursor(5, 10);
    display.println("Calibrando...");
    display.setCursor(5, 25);
    display.print("Lectura ");
    display.print(validReadings);
    display.print("/");
    display.println(MIN_VALID_READINGS);
    display.setCursor(5, 40);
    display.println("Manten el dedo");
    display.setCursor(5, 52);
    display.println("quieto");
    display.display();

    Serial.print("📊 Calibrando... ");
    Serial.print(validReadings);
    Serial.print("/");
    Serial.println(MIN_VALID_READINGS);
  }
  sumredrms = 0.0;
  sumirrms = 0.0;
  i = 0;
 }
 i++;
}
```

A.3 RESULTADOS

A continuación, se presentan los resultados obtenidos de las pruebas comparativas realizadas entre el pulsioxímetro diseñado (N) y un pulsioxímetro comercial (C).

Los resultados están organizados en quince tablas, cada una correspondiente a un adulto distinto. Para asegurar la representatividad, cada tabla incluye 59 mediciones simultáneas por sujeto.

Las columnas que se detallan en cada tabla son:

- BPM N: Valor de la frecuencia cardiaca (en BPM) medido por el dispositivo diseñado.
- BPM C: Valor de la frecuencia cardiaca (en BPM) medido por el dispositivo comercial.
- SPO2 N: Valor de la saturación de oxígeno en la sangre (en %) medido por el dispositivo diseñado.
- SPO2 C: Valor de la saturación de oxígeno en la sangre (en %) medido por el dispositivo comercial.

- DIF BPM: Diferencia entre las mediciones de la frecuencia cardiaca de ambos dispositivos (BPM N - BPM C).
- DIF SPO2: Diferencia entre las mediciones de la saturación de oxígeno en la sangre de ambos dispositivos (SPO2 N - SPO2 C).

Tabla 14, Mediciones experimentales del sujeto de prueba 15. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
66	70	97	94	-4	3
67	72	97	94	-5	3
69	72	97	94	-3	3
72	73	97	97	-1	0
72	72	97	94	0	3
72	75	97	94	-3	3
70	77	97	94	-7	3
70	79	97	94	-9	3
74	80	97	95	-6	2
78	80	97	95	-2	2
78	77	97	95	1	2
77	76	96	95	1	1
77	74	96	94	3	2
75	74	96	94	1	2
74	75	96	94	-1	2
74	77	96	94	-3	2
74	79	96	94	-5	2
79	80	96	94	-1	2
80	79	96	94	1	2
80	78	96	94	2	2
76	78	96	94	-2	2
76	77	96	94	-1	2
76	77	96	94	-1	2
85	77	96	94	8	2
87	79	96	94	8	2
78	78	96	94	0	2
78	78	96	93	0	3

86	78	96	93	8	3
86	79	96	93	7	3
89	83	96	93	6	3
89	84	96	93	5	3
86	84	96	93	2	3
86	85	96	93	1	3
93	84	96	93	9	3
90	84	96	93	6	3
90	84	96	94	6	2
79	81	96	94	-2	2
79	79	96	94	0	2
79	78	96	95	1	1
79	77	96	95	2	1
79	76	96	95	3	1
77	76	96	96	1	0
77	78	96	96	-1	0
79	78	96	96	1	0
79	77	96	96	2	0
76	76	96	96	0	0
77	75	97	96	2	1
77	74	97	95	3	2
77	75	97	95	2	2
76	76	97	95	0	2
76	77	97	95	-1	2
76	78	97	95	-2	2
81	77	97	95	4	2
80	77	97	95	3	2
80	77	97	94	3	3
77	77	97	94	0	3
77	79	97	94	-2	3
77	80	97	95	-3	2
78	79	97	95	-1	2

Tabla 15. Mediciones experimentales del sujeto de prueba 14. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
87	77	76	95	10	-19
80	78	81	95	2	-14
76	78	85	95	-2	-10
76	76	85	95	0	-10
78	77	89	95	1	-6
80	80	91	95	0	-4
81	82	93	95	-1	-2
81	83	93	95	-2	-2
83	86	96	95	-3	1
84	85	96	95	-1	1
84	84	96	95	0	1
89	82	97	95	7	2
84	78	98	95	6	3
75	78	98	95	-3	3
76	79	98	95	-3	3
79	80	98	95	-1	3
79	81	98	95	-2	3
81	81	98	95	0	3
81	82	98	95	-1	3
81	81	98	95	0	3
86	82	98	95	4	3
86	83	98	95	3	3
82	83	98	95	-1	3
82	84	98	95	-2	3
81	83	98	95	-2	3
81	83	98	95	-2	3
81	84	98	95	-3	3
82	84	98	95	-2	3
82	85	98	95	-3	3
83	86	98	95	-3	3
83	85	98	95	-2	3
83	84	98	95	-1	3
83	82	98	95	1	3
83	80	98	95	3	3

83	79	98	95	4	3
83	80	98	95	3	3
83	78	98	95	5	3
79	79	98	95	0	3
79	78	98	95	1	3
79	77	98	95	2	3
79	78	98	95	1	3
79	79	98	95	0	3
79	80	98	95	-1	3
79	81	98	95	-2	3
79	82	98	95	-3	3
79	83	98	95	-4	3
79	82	98	95	-3	3
79	81	98	95	-2	3
79	79	98	95	0	3
79	78	98	95	1	3
79	79	98	95	0	3
78	79	98	95	-1	3
78	80	98	95	-2	3
78	81	98	95	-3	3
78	82	98	95	-4	3
78	83	98	95	-5	3
81	82	98	95	-1	3
81	81	98	95	0	3
81	80	98	95	1	3

Tabla 16, Mediciones experimentales del sujeto de prueba 13. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
80	70	82	95	10	-13
83	71	85	95	12	-10
83	70	85	95	13	-10
83	68	87	95	15	-8
83	67	87	95	16	-8
73	65	88	95	8	-7
66	64	89	95	2	-6
66	64	89	96	2	-7
66	64	88	95	2	-7
66	66	88	95	0	-7
66	67	88	95	-1	-7
71	68	88	95	3	-7
70	68	89	95	2	-6
70	66	89	95	4	-6
70	65	89	95	5	-6
70	64	89	95	6	-6
70	64	87	95	6	-8
70	66	85	95	4	-10
70	67	85	95	3	-10
70	68	84	95	2	-11
70	71	84	95	-1	-11
70	72	84	96	-2	-12
70	72	85	96	-2	-11
70	70	87	96	0	-9
70	68	87	96	2	-9
70	66	87	96	4	-9
70	69	88	96	1	-8
75	70	89	96	5	-7
73	75	90	95	-2	-5
73	77	90	95	-4	-5
73	75	91	95	-2	-4

74	78	92	95	-4	-3
74	76	92	95	-2	-3
74	78	92	95	-4	-3
74	77	92	95	-3	-3
78	76	92	95	2	-3
77	76	93	95	1	-2
77	73	93	94	4	-1
77	72	93	94	5	-1
74	70	94	94	4	0
74	68	94	94	6	0
71	67	94	94	4	0
66	66	95	94	0	1
66	64	95	95	2	0
65	65	95	95	0	0
65	66	95	96	-1	-1
64	66	95	96	-2	-1
64	72	95	96	-8	-1
64	73	95	96	-9	-1
64	71	95	96	-7	-1
64	66	94	96	-2	-2
64	68	94	96	-4	-2
65	68	94	96	-3	-2
64	66	94	96	-2	-2
67	70	94	96	-3	-2
67	72	94	97	-5	-3
71	79	94	97	-8	-3
76	78	93	97	-2	-4
73	76	94	97	-3	-3

Tabla 17, Mediciones experimentales del sujeto de prueba 12. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
76	70	70	90	6	-20
76	67	70	90	9	-20
76	66	76	90	10	-14
76	64	76	90	12	-14
66	64	80	91	2	-11
67	64	83	91	3	-8
67	65	83	91	2	-8
67	64	83	91	3	-8
64	65	86	92	-1	-6
65	65	87	93	0	-6
65	64	87	93	1	-6
65	65	89	93	0	-4
65	65	89	93	0	-4
65	63	89	94	2	-5
70	64	89	94	6	-5
70	65	89	94	5	-5
70	68	89	94	2	-5
70	70	89	94	0	-5
70	74	89	94	-4	-5
71	78	89	95	-7	-6
66	75	90	95	-9	-5
68	75	90	95	-7	-5
68	69	90	94	-1	-4
66	67	90	94	-1	-4
65	67	91	93	-2	-2
70	67	91	93	3	-2
70	66	91	93	4	-2
70	68	91	93	2	-2
70	69	91	93	1	-2
69	68	92	94	1	-2
69	66	92	94	3	-2
71	65	92	94	6	-2
71	65	91	94	6	-3
71	67	91	94	4	-3

71	66	91	94	5	-3
76	67	91	94	9	-3
76	66	91	94	10	-3
75	65	91	94	10	-3
75	66	91	94	9	-3
75	67	91	94	8	-3
75	69	91	94	6	-3
88	70	91	95	18	-4
80	76	92	93	4	-1
80	73	92	93	7	-1
80	72	92	93	8	-1
90	76	92	93	14	-1
83	75	93	94	8	-1
76	74	93	94	2	-1
76	74	93	94	2	-1
76	76	93	94	0	-1
76	77	93	94	-1	-1
76	75	93	94	1	-1
69	73	94	94	-4	0
69	71	94	94	-2	0
64	76	94	94	-12	0
64	78	94	95	-14	-1
64	77	93	94	-13	-1
67	78	93	94	-11	-1
67	78	93	94	-11	-1

Tabla 18, Mediciones experimentales del sujeto de prueba 11. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
84	85	86	95	-1	-9
84	86	86	95	-2	-9
84	87	86	95	-3	-9
84	85	89	95	-1	-6
84	84	91	96	0	-5
84	83	91	96	1	-5
83	82	93	96	1	-3
83	80	93	95	3	-2
83	80	93	95	3	-2
83	80	94	95	3	-1
83	81	94	95	2	-1
81	81	95	95	0	0
82	81	96	95	1	1
82	80	96	95	2	1
82	78	96	94	4	2
82	79	96	94	3	2
81	80	96	94	1	2
81	81	96	94	0	2
81	82	96	94	-1	2
81	81	96	94	0	2
81	80	96	94	1	2
81	78	96	94	3	2
81	79	95	94	2	1
81	81	94	94	0	0
81	82	94	94	-1	0
78	83	94	94	-5	0
78	82	94	94	-4	0
78	81	94	94	-3	0
78	80	94	94	-2	0
79	79	94	94	0	0
78	80	94	94	-2	0
78	81	94	94	-3	0
78	83	94	94	-5	0
95	86	95	94	9	1

96	84	95	94	12	1
82	84	95	94	-2	1
82	82	95	94	0	1
86	81	96	94	5	2
86	82	96	94	4	2
86	81	96	94	5	2
79	81	96	94	-2	2
79	79	97	94	0	3
73	79	97	94	-6	3
73	78	97	94	-5	3
73	79	97	94	-6	3
73	82	97	95	-9	2
83	81	97	95	2	2
81	80	97	95	1	2
81	79	97	95	2	2
81	78	97	95	3	2
81	76	97	95	5	2
79	78	98	95	1	3
79	84	98	95	-5	3
84	84	98	95	0	3
83	84	98	95	-1	3
83	81	98	95	2	3
83	79	98	95	4	3
82	80	98	94	2	4
82	81	98	94	1	4

Tabla 19, Mediciones experimentales del sujeto de prueba 10. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
76	87	91	95	-11	-4
76	87	91	95	-11	-4
70	85	92	96	-15	-4
62	86	92	95	-24	-3
77	88	93	95	-11	-2
77	87	93	96	-10	-3
84	83	93	96	1	-3
84	81	83	86	3	-3
82	82	93	95	0	-2
64	81	94	95	-17	-1
64	82	94	95	-18	-1
64	81	94	96	-17	-2
65	76	94	96	-11	-2
73	75	95	96	-2	-1
83	76	95	96	7	-1
83	81	95	96	2	-1
83	83	95	96	0	-1
83	84	95	96	-1	-1
84	84	95	96	0	-1
79	83	95	96	-4	-1
82	83	96	96	-1	0
84	82	96	96	2	0
84	81	96	97	3	-1
83	80	96	97	3	-1
83	79	97	97	4	0
83	75	97	97	8	0
90	74	97	97	16	0
88	73	97	97	15	0
78	73	97	97	5	0
78	74	97	97	4	0
78	72	97	97	6	0
76	73	97	97	3	0
70	72	97	97	-2	0
76	74	98	97	2	1

70	78	97	97	-8	0
68	78	97	97	-10	0
68	80	97	97	-12	0
68	82	97	97	-14	0
70	83	97	97	-13	0
84	80	97	97	4	0
84	79	97	97	5	0
82	79	97	97	3	0
82	81	97	97	1	0
82	80	97	97	2	0
82	81	97	97	1	0
81	79	98	97	2	1
81	80	98	96	1	2
81	81	98	96	0	2
81	80	98	96	1	2
81	79	98	96	2	2
81	78	97	96	3	1
86	78	97	96	8	1
86	82	97	97	4	0
86	86	97	97	0	0
84	88	94	97	-4	-3
84	84	93	97	0	-4
84	83	93	97	1	-4
84	85	93	97	-1	-4
84	84	93	96	0	-3

Tabla 20, Mediciones experimentales del sujeto de prueba 9. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
108	110	70	91	-2	-21
108	109	70	91	-1	-21
109	110	75	91	-1	-16
111	111	79	91	0	-12
111	111	82	91	0	-9
111	111	82	92	0	-10
112	113	84	92	-1	-8
109	111	87	92	-2	-5
109	108	87	92	1	-5
107	105	88	92	2	-4
107	110	88	92	-3	-4
108	113	90	92	-5	-2
109	113	91	92	-4	-1
102	104	92	92	-2	0
102	111	92	92	-9	0
104	115	94	92	-11	2
111	120	95	92	-9	3
111	111	95	92	0	3
111	105	95	93	6	2
111	100	95	93	11	2
118	102	95	93	16	2
103	104	95	93	-1	2
103	107	95	93	-4	2
93	109	96	93	-16	3
109	116	96	93	-7	3
109	108	96	93	1	3
98	97	96	93	1	3
98	94	96	93	4	3
98	95	96	93	3	3
98	97	96	93	1	3
98	101	96	93	-3	3
98	103	96	93	-5	3
96	103	97	92	-7	5
107	108	97	92	-1	5

107	107	97	92	0	5
107	107	97	92	0	5
107	108	97	92	-1	5
109	108	97	93	1	4
109	110	97	93	-1	4
100	110	97	93	-10	4
98	104	97	93	-6	4
97	104	97	93	-7	4
97	105	97	93	-8	4
98	107	97	93	-9	4
116	110	97	93	6	4
116	112	97	93	4	4
116	111	97	93	5	4
116	110	97	93	6	4
110	109	97	93	1	4
110	109	97	93	1	4
113	111	97	93	2	4
113	118	97	93	-5	4
120	118	96	93	2	3
120	120	96	93	0	3
120	116	96	93	4	3
120	113	96	93	7	3
118	111	96	93	7	3
118	112	96	93	6	3
114	113	96	94	1	2

Tabla 21, Mediciones experimentales del sujeto de prueba 8. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
63	56	70	91	7	-21
63	59	70	91	4	-21
63	67	70	91	-4	-21
63	67	70	92	-4	-22
63	68	75	92	-5	-17
63	69	75	92	-6	-17
66	68	84	92	-2	-8
66	67	84	92	-1	-8
66	66	85	93	0	-8
66	66	85	94	0	-9
73	67	86	94	6	-8
72	68	87	94	4	-7
72	69	87	94	3	-7
72	68	87	94	4	-7
78	68	87	93	10	-6
76	69	87	93	7	-6
70	69	88	93	1	-5
70	69	88	92	1	-4
70	69	88	92	1	-4
70	68	88	93	2	-5
72	68	89	93	4	-4
70	68	90	94	2	-4
69	67	90	94	2	-4
70	67	91	94	3	-3
68	67	92	94	1	-2
68	68	92	94	0	-2
68	68	92	94	0	-2
68	68	93	94	0	-1
68	69	93	94	-1	-1
80	72	93	94	8	-1
80	71	93	94	9	-1

86	70	93	94	16	-1
76	67	94	94	9	0
76	68	94	94	8	0
76	69	94	94	7	0
76	70	94	94	6	0
70	67	94	94	3	0
67	67	95	94	0	1
67	66	95	94	1	1
66	65	96	94	1	2
66	66	96	94	0	2
64	67	95	95	-3	0
64	69	95	95	-5	0
62	69	95	95	-7	0
62	70	95	95	-8	0
71	69	95	95	2	0
71	68	95	94	3	1
70	68	96	94	2	2
70	69	96	94	1	2
72	69	96	94	3	2
72	70	96	94	2	2
72	69	96	94	3	2
72	69	95	93	3	2
72	69	95	93	3	2
76	70	95	93	6	2
76	74	95	93	2	2
76	76	95	93	0	2
76	78	95	93	-2	2
76	81	95	93	-5	2

Tabla 22, Mediciones experimentales del sujeto de prueba 7. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
93	82	93	94	11	-1
102	100	94	94	2	0
98	102	95	94	-4	1
98	99	95	94	-1	1
98	98	95	94	0	1
98	96	95	95	2	0
96	97	96	95	-1	1
97	98	96	95	-1	1
97	101	96	95	-4	1
105	105	96	94	0	2
105	104	96	94	1	2
105	103	96	94	2	2
98	102	96	94	-4	2
98	101	96	94	-3	2
95	100	96	94	-5	2
101	101	96	94	0	2
101	102	96	94	-1	2
96	104	97	94	-8	3
96	105	97	94	-9	3
106	105	97	94	1	3
107	103	97	94	4	3
107	100	97	94	7	3
107	99	97	94	8	3
101	101	97	94	0	3
103	103	97	94	0	3
103	102	97	94	1	3
103	99	97	94	4	3
103	96	97	95	7	2
102	94	97	95	8	2
99	95	97	95	4	2
99	96	97	95	3	2
99	99	97	95	0	2

99	103	97	95	-4	2
99	106	97	95	-7	2
103	105	97	95	-2	2
103	101	97	94	2	3
103	100	97	94	3	3
103	98	97	94	5	3
103	97	97	94	6	3
97	101	97	94	-4	3
101	103	97	94	-2	3
103	102	97	94	1	3
103	99	97	94	4	3
103	98	97	94	5	3
103	96	97	94	7	3
103	95	97	94	8	3
99	92	97	94	7	3
91	94	96	94	-3	2
91	96	96	94	-5	2
98	104	96	94	-6	2
98	102	96	94	-4	2
96	98	96	94	-2	2
96	95	96	94	1	2
93	88	96	94	5	2
83	81	96	94	2	2
82	85	96	94	-3	2
97	95	96	94	2	2
98	94	96	93	4	3
98	95	96	93	3	3

Tabla 23, Mediciones experimentales del sujeto de prueba 6. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
71	71	83	95	0	-12
71	70	83	95	1	-12
71	71	83	95	0	-12
71	72	83	95	-1	-12
71	72	84	95	-1	-11
71	71	87	95	0	-8
71	72	87	95	-1	-8
71	72	88	96	-1	-8
71	74	88	96	-3	-8
71	76	88	96	-5	-8
69	76	89	96	-7	-7
69	75	89	96	-6	-7
77	75	89	96	2	-7
77	75	89	96	2	-7
77	74	89	96	3	-7
79	73	91	96	6	-5
73	74	92	96	-1	-4
73	76	92	96	-3	-4
79	76	93	97	3	-4
79	77	93	97	2	-4
79	76	93	96	3	-3
82	75	94	96	7	-2
81	75	94	96	6	-2
76	75	95	96	1	-1
75	75	95	96	0	-1
75	75	95	96	0	-1
74	75	97	96	-1	1
74	75	97	96	-1	1
74	76	97	96	-2	1
74	78	97	96	-4	1
74	75	97	96	-1	1
76	79	98	96	-3	2
76	78	98	96	-2	2
77	78	98	97	-1	1

77	79	98	97	-2	1
76	77	98	97	-1	1
76	76	98	96	0	2
73	76	98	96	-3	2
84	75	97	96	9	1
84	77	97	96	7	1
84	79	97	96	5	1
84	90	97	96	-6	1
84	81	97	96	3	1
84	82	98	96	2	2
84	83	98	96	1	2
82	82	98	96	0	2
81	82	98	96	-1	2
81	81	98	96	0	2
81	80	98	96	1	2
81	81	98	96	0	2
83	83	98	96	0	2
83	85	98	96	-2	2
84	86	98	96	-2	2
81	85	98	96	-4	2
81	83	98	96	-2	2
81	81	98	96	0	2
81	79	98	96	2	2
76	78	98	96	-2	2
76	64	98	96	12	2

Tabla 24, Mediciones experimentales del sujeto de prueba 5. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
87	81	90	96	6	-6
93	83	91	96	10	-5
93	86	91	96	7	-5
91	87	92	96	4	-4
87	88	93	96	-1	-3
87	88	93	96	-1	-3
82	87	94	96	-5	-2
82	86	94	96	-4	-2
84	87	95	96	-3	-1
89	88	95	96	1	-1
89	89	95	96	0	-1
89	90	95	96	-1	-1
92	87	94	95	5	-1
93	89	94	95	4	-1
93	90	94	95	3	-1
93	91	94	95	2	-1
93	90	94	96	3	-2
93	89	94	96	4	-2
97	90	93	96	7	-3
97	91	93	96	6	-3
97	92	93	96	5	-3
100	92	93	96	8	-3
96	91	94	95	5	-1
93	90	94	95	3	-1
93	91	94	95	2	-1
93	92	94	95	1	-1
93	94	94	95	-1	-1
95	94	94	95	1	-1
95	95	94	95	0	-1
91	94	94	95	-3	-1
91	93	94	95	-2	-1
93	93	94	95	0	-1
93	93	94	95	0	-1
93	94	94	96	-1	-2

94	95	94	96	-1	-2
94	96	94	96	-2	-2
87	96	94	96	-9	-2
86	95	93	96	-9	-3
94	94	93	96	0	-3
94	93	93	96	1	-3
94	90	93	96	4	-3
94	89	93	96	5	-3
88	87	93	95	1	-2
87	88	93	95	-1	-2
83	88	94	95	-5	-1
89	90	94	95	-1	-1
89	89	94	95	0	-1
90	89	94	94	1	0
89	89	95	94	0	1
89	90	95	94	-1	1
91	90	95	94	1	1
91	91	95	95	0	0
91	91	95	95	0	0
91	92	95	95	-1	0
91	93	95	95	-2	0
92	92	95	95	0	0
92	91	95	95	1	0
96	89	95	95	7	0
96	88	95	95	8	0

Tabla 25, Mediciones experimentales del sujeto de prueba 4. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
71	73	70	95	-2	-25
73	73	72	95	0	-23
73	74	72	95	-1	-23
76	75	76	95	1	-19
76	76	80	95	0	-15
76	80	80	95	-4	-15
76	82	80	95	-6	-15
87	84	85	95	3	-10
87	82	85	95	5	-10
87	80	88	94	7	-6
87	78	88	94	9	-6
78	74	91	93	4	-2
78	75	91	93	3	-2
75	74	93	93	1	0
75	73	93	93	2	0
75	72	93	93	3	0
78	69	95	93	9	2
78	70	95	93	8	2
73	73	96	93	0	3
73	74	96	93	-1	3
73	74	96	93	-1	3
73	80	97	94	-7	3
80	81	97	95	-1	2
80	78	97	95	2	2
79	78	96	95	1	1
77	79	96	95	-2	1
77	78	96	95	-1	1
76	77	96	95	-1	1
76	73	96	95	3	1
74	73	96	94	1	2
74	72	96	94	2	2

69	72	96	95	-3	1
69	72	96	95	-3	1
69	72	96	95	-3	1
73	74	96	95	-1	1
73	76	96	95	-3	1
79	78	97	95	1	2
79	81	97	95	-2	2
83	81	97	96	2	1
89	91	97	95	-2	2
89	82	97	95	7	2
89	83	97	95	6	2
95	88	97	95	7	2
91	88	97	95	3	2
91	88	97	95	3	2
85	93	97	95	-8	2
85	92	97	95	-7	2
85	89	97	95	-4	2
85	86	97	95	-1	2
85	82	97	95	3	2
85	80	97	95	5	2
85	79	97	95	6	2
85	77	97	95	8	2
85	75	97	95	10	2
85	76	97	95	9	2
85	77	97	95	8	2
85	79	97	95	6	2
85	80	97	95	5	2
85	81	97	95	4	2

Tabla 26, Mediciones experimentales del sujeto de prueba 3. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
76	72	70	96	4	-26
76	70	70	96	6	-26
76	67	70	96	9	-26
70	67	78	96	3	-18
70	68	78	96	2	-18
76	67	85	96	9	-11
73	65	85	96	8	-11
73	65	88	96	8	-8
70	66	88	96	4	-8
72	66	90	96	6	-6
75	67	91	97	8	-6
75	67	91	97	8	-6
75	66	91	97	9	-6
73	63	93	98	10	-5
70	63	93	98	7	-5
70	62	93	98	8	-5
70	63	93	98	7	-5
73	62	94	99	11	-5
73	63	94	99	10	-5
70	62	93	99	8	-6
63	62	93	99	1	-6
63	64	93	99	-1	-6
66	66	92	99	0	-7
66	67	92	99	-1	-7
66	66	92	99	0	-7
74	68	92	99	6	-7
74	72	92	99	2	-7
68	69	93	99	-1	-6
68	68	93	99	0	-6
68	71	93	99	-3	-6
68	74	93	99	-6	-6
69	72	93	99	-3	-6
65	73	93	99	-8	-6
65	75	93	99	-10	-6

64	74	94	99	-10	-5
76	75	94	99	1	-5
76	77	94	99	-1	-5
73	77	94	99	-4	-5
73	75	94	99	-2	-5
73	76	94	99	-3	-5
86	79	95	99	7	-4
78	80	95	99	-2	-4
75	79	95	99	-4	-4
75	81	95	99	-6	-4
83	80	96	99	3	-3
83	79	96	99	4	-3
78	79	96	99	-1	-3
78	77	96	99	1	-3
78	75	96	99	3	-3
78	73	96	99	5	-3
77	72	97	98	5	-1
77	71	97	98	6	-1
77	75	97	98	2	-1
77	73	97	98	4	-1
77	77	97	98	0	-1
77	79	97	98	-2	-1
80	78	97	98	2	-1
80	75	97	98	5	-1
76	78	97	98	-2	-1

Tabla 27, Mediciones experimentales del sujeto de prueba 2. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
92	93	83	95	-1	-12
92	92	83	95	0	-12
92	90	83	95	2	-12
92	89	83	95	3	-12
91	88	86	95	3	-9
91	91	86	95	0	-9
90	92	88	95	-2	-7
90	91	88	95	-1	-7
88	90	90	95	-2	-5
88	88	90	95	0	-5
88	87	90	95	1	-5
90	87	92	95	3	-3
90	89	92	95	1	-3
90	94	92	95	-4	-3
93	97	93	95	-4	-2
93	98	93	95	-5	-2
96	97	94	94	-1	0
96	94	94	94	2	0
96	94	94	94	2	0
90	94	94	94	-4	0
91	95	95	94	-4	1
91	95	95	93	-4	2
92	95	95	93	-3	2
92	98	95	93	-6	2
92	96	95	93	-4	2
92	95	95	93	-3	2
92	93	95	93	-1	2
92	91	95	94	1	1
95	90	96	94	5	2
95	89	96	94	6	2
92	89	96	94	3	2
87	89	96	95	-2	1
87	88	96	95	-1	1
87	90	96	95	-3	1

87	91	96	95	-4	1
88	90	96	95	-2	1
88	89	96	95	-1	1
91	89	96	94	2	2
95	92	96	94	3	2
95	95	96	94	0	2
95	96	96	94	-1	2
95	98	96	94	-3	2
96	97	96	94	-1	2
96	95	96	94	1	2
96	94	96	94	2	2
95	92	96	93	3	3
94	91	96	93	3	3
94	93	96	93	1	3
94	94	96	93	0	3
94	97	96	93	-3	3
95	99	96	93	-4	3
99	99	96	94	0	2
99	101	96	94	-2	2
104	101	96	94	3	2
104	100	96	94	4	2
104	98	96	94	6	2
104	96	96	94	8	2
98	95	96	94	3	2
96	99	96	94	-3	2

Tabla 28, Mediciones experimentales del sujeto de prueba 1. [Elaboración propia]

BPM N	BPM C	SPO2 N	SPO2 C	DIF BPM	DIF SPO2
68	66	75	96	2	-21
68	65	75	96	3	-21
63	64	79	96	-1	-17
63	65	79	96	-2	-17
68	65	82	96	3	-14
68	66	82	96	2	-14
68	66	82	96	2	-14
68	67	86	96	1	-10
69	66	88	95	3	-7
69	68	88	95	1	-7
69	69	88	95	0	-7
70	69	90	95	1	-5
67	69	91	95	-2	-4
67	68	91	95	-1	-4
71	68	92	94	3	-2
71	69	92	94	2	-2
69	69	93	94	0	-1
73	70	94	93	3	1
68	69	94	93	-1	1
68	67	94	93	1	1
67	67	95	93	0	2
67	66	95	93	1	2
67	66	95	93	1	2
67	65	95	93	2	2
67	64	95	94	3	1
64	65	96	94	-1	2
64	64	96	94	0	2
64	63	96	94	1	2
64	62	96	94	2	2
61	62	96	94	-1	2
61	63	96	94	-2	2
65	64	96	94	1	2
65	65	96	94	0	2
67	65	96	94	2	2

68	65	96	94	3	2
68	64	96	94	4	2
65	64	96	95	1	1
65	63	96	95	2	1
65	63	96	95	2	1
66	63	97	95	3	2
68	63	97	95	5	2
68	64	97	95	4	2
68	65	97	95	3	2
66	65	97	95	1	2
68	66	97	95	2	2
65	67	97	95	-2	2
65	68	97	95	-3	2
69	68	96	95	1	1
69	69	96	95	0	1
73	69	96	95	4	1
73	70	96	95	3	1
69	70	96	95	-1	1
67	70	96	95	-3	1
69	70	96	95	-1	1
69	69	96	95	0	1
69	69	96	95	0	1
68	70	96	95	-2	1
68	70	96	95	-2	1
66	70	95	95	-4	0