
	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 1 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

Manual Técnico para Proyectos de Software

***Implementación de una clínica estética Clinestetica  
cuidad de Popayán]  
Fecha: [15/09/2025]***


	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 2 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

## HOJA DE CONTROL

<b>Empresa Organización</b>	/ ClinEstetica		
<b>Proyecto</b>	Implementación de un sistema de información para una clínica estética de la ciudad de Popayán		
<b>Entregable</b>	Manual técnico		
<b>Autores</b>	Gómez Darwin, Pérez Liliana, Quira Constanza		
<b>Versión/Edición</b>	1	<b>Fecha Versión</b>	15-09-2025
<b>Aprobado por</b>		<b>Fecha Aprobación</b>	
		<b>Nº Total de Páginas</b>	

## REGISTRO DE CAMBIOS


<b>Versión</b>	<b>Causa del Cambio</b>	<b>Responsable del Cambio</b>	<b>Fecha del Cambio</b>

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 3 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

## CONTENIDO

### Contenido

1. Introducción.....	4
2. Objetivo. ....	4
3. Descripción General del Proyecto .....	4
4. Arquitectura del sistema .....	4
5. Herramientas utilizadas para el desarrollo. ....	8
6. Instalación y configuración .....	<b>¡Error! Marcador no definido.</b>
7. Base de datos .....	<b>¡Error! Marcador no definido.</b>
8. Desarrollo del Software .....	<b>¡Error! Marcador no definido.</b>
9. Funcionalidades del software .....	<b>¡Error! Marcador no definido.</b>
10. Interfases del sistema.....	<b>¡Error! Marcador no definido.</b>
11. Bibliografía .....	50

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 4 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

## 1. Introducción

Este manual técnico proporciona la guía para la instalación, configuración y mantenimiento del sistema de gestión de citas médicas, desarrollado con Node.js, Express, MySQL (Sequelize) y React.

Está orientado a Desarrolladores, administradores de sistemas y personal de soporte técnico. El manual incluye desde la descripción del sistema, arquitectura, instalación, base de datos, desarrollo, funcionalidades e interfaces.

## 2. Objetivo.


Describir de manera detallada el funcionamiento interno del sistema de gestión de citas médicas, orientado a facilitar la instalación, configuración y soporte.

## 3. Descripción General del Proyecto

El proyecto corresponde a un Sistema de Gestión Clínica Estética para la ciudad de Popayán que permite administrar usuarios, citas, procedimientos, historiales médicos y notificaciones en tiempo real mediante sockets. Cuenta con roles diferenciados (Doctor, Asistente, Usuario) y paneles específicos para cada rol.

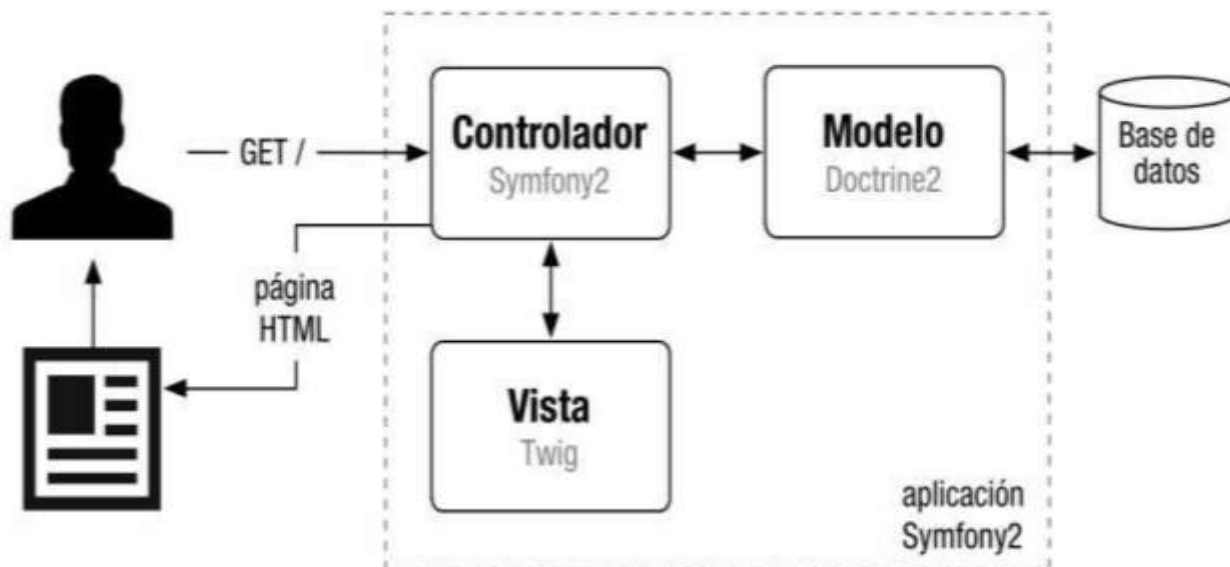
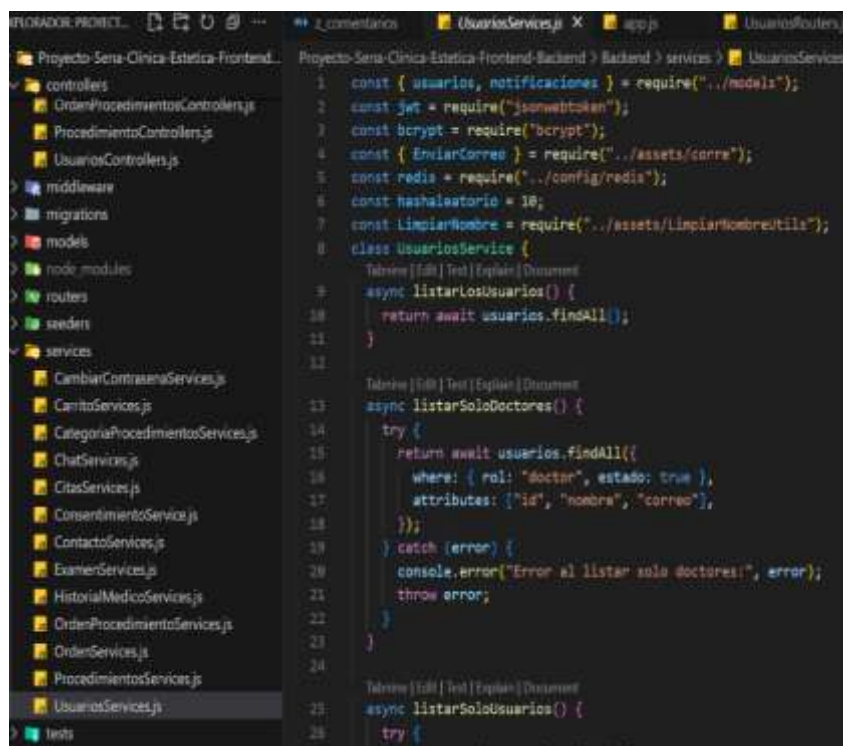
## 4. Arquitectura del sistema

El modelo MVC (Modelo, Vista, Controlador) es un esquema de arquitectura por capas muy utilizado en el desarrollo de software basado en aplicaciones web. El modelo (M) controla todo lo relacionado con los datos, la vista (V) lo relacionado con las interfaces de usuario y el controlador (C) se encarga de la manipulación del M para mostrar

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 5 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

información en la V.

## VISTA CONTROLADOR

```

Projecto-Sena-Clinica-Estetica-Frontend-Backend > Backend > services > UsuariosServices.js
1  const { usuarios, notificaciones } = require("../models");
2  const jwt = require("jsonwebtoken");
3  const bcrypt = require("bcrypt");
4  const { EnviarCorreo } = require("../assets/corre");
5  const redis = require("../config/redis");
6  const hashaleatorio = 10;
7  const LimpiarNombre = require("../assets/LimpiarNombreUtils");
8  class UsuariosService {
9      Tabview | Edit | Test | Explain | Document
10     async listarUsuarios() {
11         return await usuarios.findAll();
12     }
13     Tabview | Edit | Test | Explain | Document
14     async listarSoloDoctores() {
15         try {
16             return await usuarios.findAll({
17                 where: { rol: "doctor", estado: true },
18                 attributes: { "id", "nombre", "correo" },
19             });
20         } catch (error) {
21             console.error("Error al listar solo doctores!", error);
22             throw error;
23         }
24     }
25     Tabview | Edit | Test | Explain | Document
26     async listarSoloUsuarios() {
27         try {
28

```



Número de Documento:

FS-DOC Formato Manual Técnico


Fecha de Creación:  
01/02/2025Elaborado por:  
Grupo instructores área de software

Nombre del Documento:

Formato para la construcción del Manual Técnico

```
Proyecto-Sena-Clinica-Estetica-Frontend-Backend > Backend > controllers > UsuariosController.js
1  const usuariosService = require("../services/UsuariosServices");
2  const {
3    registrarIntentoFallido,
4    limpiarIntentos,
5  } = require("../middleware/intentosfallidos");
6  class UsuariosController {
7    // ...
8    async listarUsuarios(req, res) {
9      const usuarios = await usuariosService.listarLosUsuarios();
10     res.json(usuarios);
11   }
12   // ...
13   async buscarUsuarios(req, res) {
14     const usuario = await usuariosService.buscarLosUsuarios(req.query.usuario);
15     ? res.json(usuario)
16     : res.status(404).json({ error: "Usuario no encontrado" });
17   }
18   // ...
19   async listarDoctores(req, res) {
20     try {
21       const doctores = await usuariosService.listarSoloDoctores();
22       res.json(doctores);
23     } catch (error) {
24       console.error("Error al listar doctores:", error);
25       res.status(500).json({ error: "Error al obtener doctores" });
26     }
27   }
28 }

Proyecto-Sena-Clinica-Estetica-Frontend-Backend > Backend > routers > UsuariosRoutes.js
1  const express = require("express");
2  const router = express.Router();
3  const usuariosController = require("../controllers/UsuariosController");
4  const { authorization, verificarRol } = require("../middleware/Authorization");
5  const { validarUsuario, validarUsuarioPublico } = require("../middleware/validarUsuario");
6  const { verificarIntentos } = require("../middleware/intentosfallidos");
7
8  router.get("/perfil", authorization, usuariosController.perfil);
9  router.post("/iniciarsesion", verificarIntentos, usuariosController.iniciarSesion);
10 // Permitir que doctores y asistentes puedan listar usuarios
11 router.get("/listarusuarios", authorization, verificarRol(["doctor", "asistente"]), usuariosController.listarUsuarios);
12 router.get("/doctores", authorization, usuariosController.listarDoctores);
13 router.get("/usuarios", authorization, usuariosController.listarUsuarios);
14 router.get("/buscarusuarios/:id", usuariosController.buscarUsuario);
15 // Flujo de registro con verificación previa
16 router.post("/preregistro", validarUsuarioPublico, usuariosController.preregistro);
17 router.post("/preregistro/confirmar", usuariosController.confirmarPreregistro);
18 // Registro público: no requiere token ni rol; el backend se encarga de validar
19 router.post("/crearusuarios", validarUsuarioPublico, usuariosController.crearUsuarios);
20 router.post("/crearusuariosadmin", authorization, verificarRol(["admin"]), usuariosController.crearUsuariosAdmin);
21 router.patch("/editarusuarios/:id", authorization, usuariosController.editarUsuario);
22 router.delete("/eliminarusuarios/:id", authorization, usuariosController.eliminarUsuario);
23 router.patch("/editarstatususuario/:id", usuariosController.editarStatusUsuario);
24 router.get("/notificaciones/:id", authorization, usuariosController.listarNotificaciones);
25 router.get("/notificacionesusuario/:id", authorization, usuariosController.listarNotificacionesUsuario);
26 router.patch("/notificaciones/:id/marcar-leida", authorization, usuariosController.marcarNotificacionLeida);
27 router.patch("/notificaciones/:id/marcar-todas-leidas", authorization, usuariosController.marcarTodasNotificacionesLeidas);
28 router.patch("/notificaciones/:id/archivar-leidas", authorization, usuariosController.archivarNotificaciones);
29 router.get("/notificaciones/:id/historial", authorization, usuariosController.listarHistorialNotificaciones);
```

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 7 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

El sistema ha sido diseñado siguiendo principios de escalabilidad horizontal mediante contenedores Docker, lo que permite incrementar recursos en función de la demanda. Además, se implementan prácticas de seguridad como sanitización de entradas, uso de HTTPS y control de acceso basado en roles.


El sistema basado en una arquitectura Cliente–Servidor, donde:

- Cliente (Frontend): desarrollado en React con Vite, CoreUI y Redux. Se encarga de la interacción con el usuario, renderización de vistas y consumo de la API del servidor.
- Servidor (Backend): implementado en Node.js con Express, siguiendo el patrón de diseño Modelo–Vista–Controlador (MVC), lo que permite separar responsabilidades en:
  - Modelos (Models): definición de las entidades y su relación con la base de datos mediante Sequelize.
  - Vistas (Views): no se renderizan en el servidor, pero se entiende como la información enviada al cliente (JSON en las APIs).
  - Controladores (Controllers): lógica de negocio que recibe las solicitudes, procesa datos y responde al cliente.

Además, el sistema cuenta con:

- Base de datos relacional: MySQL, administrada con Sequelize.
- Comunicación en tiempo real: Socket.io para notificaciones instantáneas.
- Seguridad: autenticación mediante JWT, manejo de roles y cifrado de contraseñas con bcrypt.




	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 8 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

## 5. Herramientas utilizadas para el desarrollo.


### Herramientas utilizadas para el desarrollo






	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 9 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

CATEGORIA	HERRAMIENTAS Y DESCRIPCIÓN DE USO
Lenguajes, Frameworks y SDKs	<ul style="list-style-type: none"> <li>• <b>JavaScript (ES6+)</b>: Lenguaje principal del sistema, usado en el frontend (React.js) y el backend (Node.js + Express).</li> <li>• <b>Dart (Flutter 3.22+)</b>: Lenguaje empleado en la aplicación móvil para Android.</li> <li>• <b>SQL (MySQL 8.x)</b>: Lenguaje para la definición y manipulación de datos relacionales.</li> <li>• <b>SDKs</b>: <ul style="list-style-type: none"> <li>– Node.js SDK: entorno de ejecución para el backend.</li> <li>– Flutter SDK: kit de desarrollo para la app móvil.</li> <li>– Cloudinary SDK: integración para la gestión de imágenes y documentos PDF.</li> <li>– Google Maps SDK: integración para el módulo de ubicación de la clínica.</li> </ul> </li> </ul>
IDEs y utilidades	<ul style="list-style-type: none"> <li>• <b>Visual Studio Code</b>: entorno principal de desarrollo para backend y frontend, con extensiones de JavaScript, React y Sequelize.</li> <li>• <b>Android Studio</b>: entorno utilizado para ejecutar y depurar la aplicación móvil Flutter.</li> <li>• <b>MySQL Workbench</b>: herramienta para la administración de la base de datos local.</li> <li>• <b>Postman / Thunder Client</b>: pruebas de endpoints y validación de APIs.</li> <li>• <b>Git Bash / Terminal</b>: ejecución de comandos, migraciones y control de versiones.</li> </ul>

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 10 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

Control de versiones y repositorios	<ul style="list-style-type: none"> <li>• <b>Git + GitHub:</b> herramienta de control de versiones empleada para gestionar el código fuente y la colaboración del equipo mediante ramas por feature y Pull Requests.</li> <li>• <b>Repositorios del proyecto:</b> <ul style="list-style-type: none"> <li>– Frontend: <a href="https://github.com/Dstevengmz/Proyecto-Clinica-estetica-App.git">https://github.com/Dstevengmz/Proyecto-Clinica-estetica-App.git</a></li> <li>– Backend: <a href="https://github.com/Dstevengmz/Proyecto-Sena-Clinica-Eстетica-Frontend-Backend.git">https://github.com/Dstevengmz/Proyecto-Sena-Clinica-Eстетica-Frontend-Backend.git</a></li> <li>– Aplicación Móvil (Flutter):</li> </ul> </li> </ul>
Dependencias clave por capa	<p><b>Backend (Node.js + Express):</b> express, sequelize, mysql2, jsonwebtoken, bcryptjs, multer, cloudinary, nodemailer, redis, pdfkit, dotenv, jest, supertest.</p> <p><b>Frontend Web (React):</b> react, react-dom, axios, react-router-dom, bootstrap, coreui, react-icons, react-data-table-component, sweetalert2, moment-timezone.</p>
Base de datos y herramientas	<ul style="list-style-type: none"> <li>• <b>MySQL 8.x:</b> motor de base de datos relacional utilizado tanto en entorno local como en la nube.</li> <li>• <b>Sequelize ORM:</b> mapeo objeto-relacional para la gestión de modelos, migraciones y seeders.</li> <li>• <b>MySQL Workbench:</b> herramienta de diseño y administración local.</li> <li>• <b>Render / Railway:</b> servicios utilizados para alojar la base de datos MySQL y la API en la nube.</li> </ul>

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 11 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

## 6. Instalación y configuración

El proceso de instalación y configuración del sistema *Clinestetica* tiene como finalidad preparar el entorno necesario para la ejecución de la aplicación web y móvil, tanto en ambiente local como en servidor.

A continuación, se describen de manera general los requisitos de hardware y software empleados durante el desarrollo y las pruebas.

### 6.1 Requisitos del sistema

- Procesador: Intel i5 o superior
- Memoria RAM: mínimo 8 GB
- Espacio en disco: mínimo 10 GB libres
- Conectividad: conexión a internet estable

### 6.2 Requisitos de software


- Sistema operativo: Windows 10 / Linux Mint / Ubuntu 22.04
- Node.js: versión 18 o superior
- MySQL: versión 8.0
- Sequelize ORM: herramienta utilizada para la gestión de modelos, migraciones y seeders en MySQL.
- CLI de Sequelize: instalada globalmente para ejecutar comandos de migración.
- Git: instalado para el control de versiones.
- Navegador: Google Chrome o similar.
- (Opcional): Docker Desktop y Docker Compose, para la ejecución de contenedores.
- Editor recomendado: Visual Studio Code.

Nota:

La información detallada sobre la instalación paso a paso, configuración de dependencias, conexión a la base de datos y despliegue, se encuentra documentada en el Manual de Instalación Clinestetica, adjunto a este documento.

Documento de referencia:

Anexo 1 – Documento, Manual de instalación Clinestetica.

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 12 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

## 7. Base de datos

### 7.1 Elección del motor

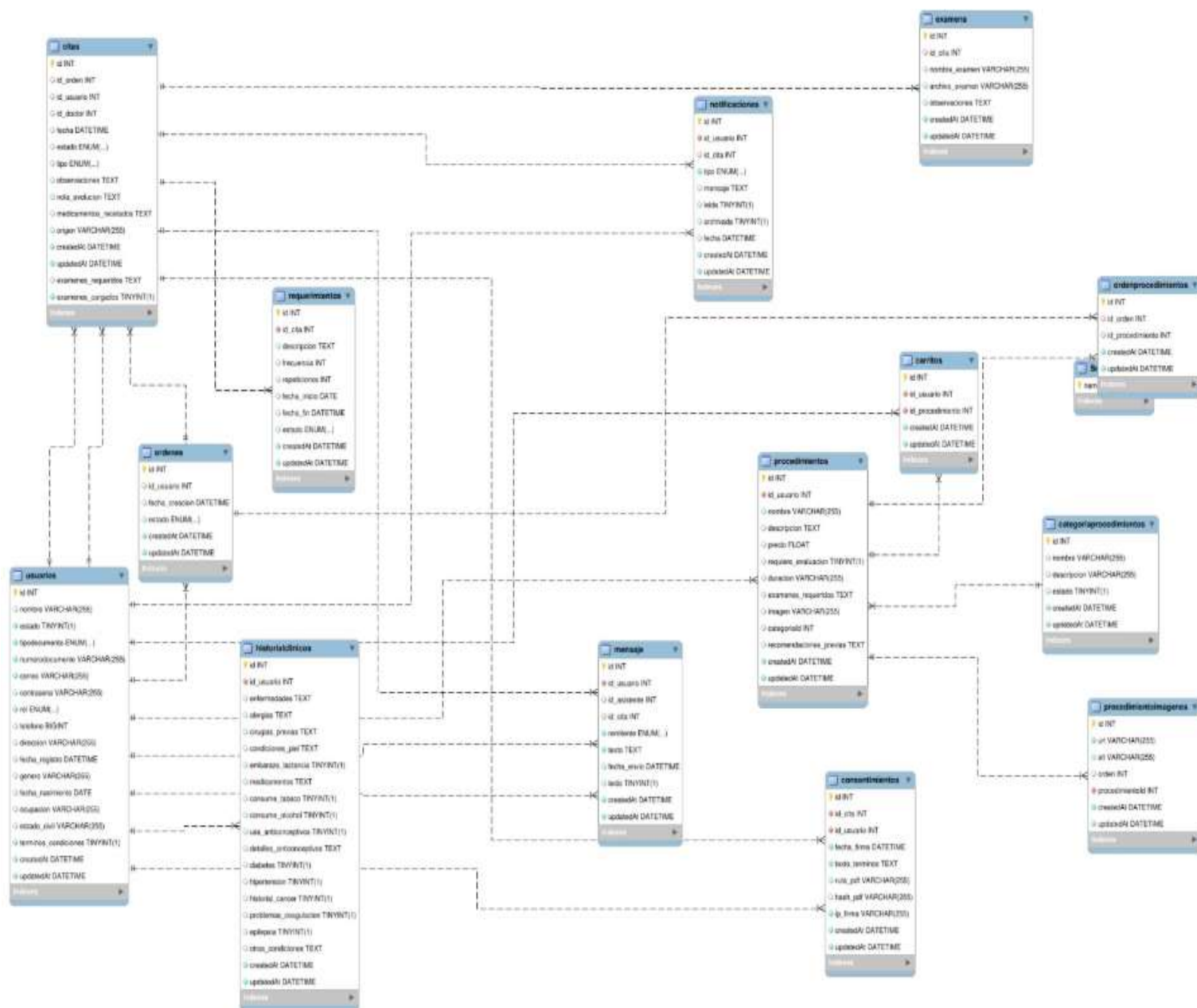
El sistema Clinestetica utiliza MySQL 8.x como motor de base de datos relacional por su rendimiento, estabilidad, escalabilidad y amplia compatibilidad con herramientas de desarrollo.

Se gestiona mediante el ORM Sequelize, que facilita el mapeo objeto-relacional y permite definir modelos, relaciones, migraciones y seeders desde el backend desarrollado en Node.js.

Como herramienta de administración local se empleó MySQL Workbench, la cual permitió realizar la ingeniería directa e inversa del esquema de datos y visualizar el modelo entidad-relación (MER).

### 7.2 Modelo entidad-relación

A continuación, se muestra el modelo entidad-relación del sistema Clinestetica, el cual describe las principales entidades y sus relaciones entre usuarios, citas, procedimientos, órdenes, consentimientos, notificaciones e historiales clínicos.



### 7.3 Diccionario de datos



	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			Página <b>14</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

TABLA	DESCRIPCION
Usuarios	Almacena los datos personales, credenciales y roles del sistema (doctor, asistente, usuario).
Procedimientos	Contiene la información de los tratamientos o procedimientos estéticos.
Categoría procedimientos	Agrupar los procedimientos por categorías.
Carrito	Registra los procedimientos seleccionados por los usuarios antes de generar una orden.
Ordenes	Representa las órdenes generadas a partir del carrito y su estado.
Orden Procedimiento	Relaciona las órdenes con los procedimientos correspondientes.
Citas	Gestiona la programación, estado y detalles de las citas médicas o estéticas.
Consentimientos	Registra la firma digital y el PDF del consentimiento informado asociado a cada cita.
Exámenes	Guarda los exámenes solicitados y cargados por los usuarios o doctores.
Historial clínico	Contiene el historial médico del usuario (enfermedades, alergias, cirugías, etc.).
Notificaciones	Almacena los avisos generados por el sistema (confirmaciones, cancelaciones, recordatorios, etc.).
Mensaje	Permite la comunicación interna entre usuario y asistente.
Requerimientos	Define tareas o indicaciones asociadas a citas o procedimientos.
Procedimiento imagen	Almacena las imágenes asociadas a cada procedimiento.

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 15 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

#### 7.4 Normalización e integridad

- Las tablas se encuentran normalizadas a la tercera forma normal (3FN), evitando redundancias y asegurando que los datos sean atómicos.
- Todas las relaciones están definidas mediante claves foráneas (FK) con reglas ON UPDATE CASCADE y ON DELETE CASCADE o SET NULL, garantizando la integridad referencial.
- Los campos createdAt y updatedAt son manejados automáticamente por Sequelize para el control de auditoría.

#### 7.5 Seguridad de datos

- Las contraseñas se almacenan encriptadas con bcrypt.
- Acceso restringido mediante roles (doctor, asistente, usuario).
- Conexiones seguras configuradas mediante variables de entorno (.env).
- Políticas de integridad aplicadas en el ORM y la base de datos.
- Copias de seguridad gestionadas desde MySQL Workbench y Render para entornos de despliegue.

#### 7.6 Script DDL de la base de datos – Clinestetica


El siguiente script fue generado automáticamente mediante la herramienta MySQL Workbench (Forward Engineering) y define la estructura completa del esquema clinicaesteticasena, incluyendo las tablas, claves primarias, foráneas, índices y relaciones entre entidades.

Codigo base de datos: -- MySQL Workbench Forward Engineering

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
```

```
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
```



	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 16 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE,  
SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

-- -----

-- Schema mydb

-- -----

-- -----

-- Schema clinicaesteticasena


-- -----

-- -----

-- Schema clinicaesteticasena

-- -----

CREATE SCHEMA IF NOT EXISTS `clinicaesteticasena` DEFAULT CHARACTER SET

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 17 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci ;

USE `clinicaesteticasena` ;

-- -----

-- Table `clinicaesteticasena`.`SequelizeMeta`

-- -----

CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`SequelizeMeta` (

  `name` VARCHAR(255) COLLATE 'utf8mb3\_unicode\_ci' NOT NULL,

  PRIMARY KEY (`name`),

  UNIQUE INDEX `name` (`name` ASC) VISIBLE)

ENGINE = InnoDB;

-- -----

-- Table `clinicaesteticasena`.`usuarios`

-- -----


CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`usuarios` (

  `id` INT NOT NULL AUTO\_INCREMENT,

  `nombre` VARCHAR(255) NULL DEFAULT NULL,

  `estado` TINYINT(1) NOT NULL DEFAULT '1',

  `tipodocumento` ENUM('Cédula de Ciudadanía', 'Pasaporte', 'Documento de

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 18 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

```

Identificación Extranjero', 'Permiso Especial de Permanencia') NOT NULL,

`numerodocumento` VARCHAR(255) NOT NULL,

`correo` VARCHAR(255) NOT NULL,

`contrasena` VARCHAR(255) NULL DEFAULT NULL,

`rol` ENUM('doctor', 'usuario', 'asistente') NOT NULL,

`telefono` BIGINT NULL DEFAULT NULL,

`direccion` VARCHAR(255) NULL DEFAULT NULL,

`fecha_registro` DATETIME NULL DEFAULT NULL,

`genero` VARCHAR(255) NULL DEFAULT NULL,

`fecha_nacimiento` DATE NULL DEFAULT NULL,

`ocupacion` VARCHAR(255) NULL DEFAULT NULL,

`estado_civil` VARCHAR(255) NULL DEFAULT NULL,

`terminos_condiciones` TINYINT(1) NOT NULL DEFAULT '0',

`createdAt` DATETIME NOT NULL,

`updatedAt` DATETIME NOT NULL,

PRIMARY KEY (`id`),

UNIQUE INDEX `numerodocumento` (`numerodocumento` ASC) VISIBLE,


UNIQUE INDEX `correo` (`correo` ASC) VISIBLE)

ENGINE = InnoDB

AUTO_INCREMENT = 9

DEFAULT CHARACTER SET = utf8mb4

```

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 19 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

COLLATE = utf8mb4\_0900\_ai\_ci;

-- -----

-- Table `clinicaesteticasena`.`categoriaprocedimientos`

-- -----

CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`categoriaprocedimientos` (

  `id` INT NOT NULL AUTO\_INCREMENT,

  `nombre` VARCHAR(255) NULL DEFAULT NULL,

  `descripcion` VARCHAR(255) NULL DEFAULT NULL,

  `estado` TINYINT(1) NULL DEFAULT NULL,

  `createdAt` DATETIME NOT NULL,

  `updatedAt` DATETIME NOT NULL,

  PRIMARY KEY (`id`))


ENGINE = InnoDB

AUTO\_INCREMENT = 2

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4\_0900\_ai\_ci;

-- -----

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 20 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

-- Table `clinicaesteticasena`.`procedimientos`

-----

CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`procedimientos` (

  `id` INT NOT NULL AUTO\_INCREMENT,

  `id\_usuario` INT NOT NULL,

  `nombre` VARCHAR(255) NULL DEFAULT NULL,

  `descripcion` TEXT NULL DEFAULT NULL,

  `precio` FLOAT NULL DEFAULT NULL,

  `requiere\_evaluacion` TINYINT(1) NULL DEFAULT NULL,

  `duracion` VARCHAR(255) NULL DEFAULT NULL,

  `exámenes\_requeridos` TEXT NULL DEFAULT NULL,

  `imagen` VARCHAR(255) NULL DEFAULT NULL,

  `categoriald` INT NULL DEFAULT NULL,

  `recomendaciones\_previas` TEXT NULL DEFAULT NULL,

  `createdAt` DATETIME NOT NULL,

  `updatedAt` DATETIME NOT NULL,


PRIMARY KEY (`id`),

INDEX `id\_usuario` (`id\_usuario` ASC) VISIBLE,

INDEX `categoriald` (`categoriald` ASC) VISIBLE,

CONSTRAINT `procedimientos\_ibfk\_1`

FOREIGN KEY (`id\_usuario`)

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 21 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

```

REFERENCES `clinicaesteticasena`.`usuarios` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `procedimientos_ibfk_2`

FOREIGN KEY (`categoriald`)

REFERENCES `clinicaesteticasena`.`categoriaprocedimientos` (`id`)

ON DELETE SET NULL

ON UPDATE CASCADE)

ENGINE = InnoDB

AUTO_INCREMENT = 10

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4_0900_ai_ci;

```

```
-- -----
```

```
-- Table `clinicaesteticasena`.`carritos`
```

```
-- -----
```

```


CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`carritos` (

`id` INT NOT NULL AUTO_INCREMENT,

`id_usuario` INT NOT NULL,

`id_procedimiento` INT NOT NULL,

```

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 22 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

```

`createdAt` DATETIME NOT NULL,

`updatedAt` DATETIME NOT NULL,

PRIMARY KEY (`id`),

INDEX `id_usuario` (`id_usuario` ASC) VISIBLE,

INDEX `id_procedimiento` (`id_procedimiento` ASC) VISIBLE,

CONSTRAINT `carritos_ibfk_1`

FOREIGN KEY (`id_usuario`)

REFERENCES `clinicaesteticasena`.`usuarios` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `carritos_ibfk_2`

FOREIGN KEY (`id_procedimiento`)

REFERENCES `clinicaesteticasena`.`procedimientos` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB


AUTO_INCREMENT = 7

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4_0900_ai_ci;

```




	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página <b>23</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

-- -----

-- Table `clinicaesteticasena`.`ordenes`

-- -----

```
CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`ordenes` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `id_usuario` INT NULL DEFAULT NULL,
  `fecha_creacion` DATETIME NULL DEFAULT NULL,
  `estado` ENUM('pendiente', 'confirmada', 'cancelada') NULL DEFAULT NULL,
  `createdAt` DATETIME NOT NULL,
  `updatedAt` DATETIME NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `id_usuario` (`id_usuario` ASC) VISIBLE,
  CONSTRAINT `ordenes_ibfk_1`
    FOREIGN KEY (`id_usuario`)
      REFERENCES `clinicaesteticasena`.`usuarios` (`id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
AUTO_INCREMENT = 4
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```


	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página <b>24</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

-----

-- Table `clinicaesteticasena`.`citas`

-----

```
CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`citas` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `id_orden` INT NULL DEFAULT NULL,
  `id_usuario` INT NULL DEFAULT NULL,
  `id_doctor` INT NULL DEFAULT NULL,
  `fecha` DATETIME NULL DEFAULT NULL,
  `estado` ENUM('pendiente', 'confirmada', 'realizada', 'cancelada') NULL DEFAULT
```

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 25 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

NULL,

`tipo` ENUM('evaluacion', 'procedimiento') NULL DEFAULT NULL,

`observaciones` TEXT NULL DEFAULT NULL,

`nota\_evolucion` TEXT NULL DEFAULT NULL,

`medicamentos\_recetados` TEXT NULL DEFAULT NULL,

`origen` VARCHAR(255) NULL DEFAULT NULL,

`createdAt` DATETIME NOT NULL,

`updatedAt` DATETIME NOT NULL,

`exámenes\_requeridos` TEXT NULL DEFAULT NULL,

`exámenes\_cargados` TINYINT(1) NOT NULL DEFAULT '0',

PRIMARY KEY (`id`),

INDEX `id\_orden` (`id\_orden` ASC) VISIBLE,

INDEX `id\_usuario` (`id\_usuario` ASC) VISIBLE,

INDEX `id\_doctor` (`id\_doctor` ASC) VISIBLE,

CONSTRAINT `citas\_ibfk\_1`

FOREIGN KEY (`id\_orden`)


REFERENCES `clinicaesteticasena`.`ordenes` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `citas\_ibfk\_2`

FOREIGN KEY (`id\_usuario`)

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 26 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

```

REFERENCES `clinicaesteticasena`.`usuarios` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `citas_ibfk_3`

FOREIGN KEY (`id_doctor`)

REFERENCES `clinicaesteticasena`.`usuarios` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB


AUTO_INCREMENT = 10

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4_0900_ai_ci;

--
-- Table `clinicaesteticasena`.`consentimientos`
--
CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`consentimientos` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `id_cita` INT NOT NULL,
  `id_usuario` INT NOT NULL,

```

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 27 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

```

`fecha_firma` DATETIME NOT NULL,

`texto_terminos` TEXT NOT NULL,

`ruta_pdf` VARCHAR(255) NULL DEFAULT NULL,

`hash_pdf` VARCHAR(255) NULL DEFAULT NULL,

`ip_firma` VARCHAR(255) NOT NULL,

`createdAt` DATETIME NOT NULL,

`updatedAt` DATETIME NOT NULL,

PRIMARY KEY (`id`),

INDEX `id_cita` (`id_cita` ASC) VISIBLE,

INDEX `id_usuario` (`id_usuario` ASC) VISIBLE,

CONSTRAINT `consentimientos_ibfk_1`

FOREIGN KEY (`id_cita`)

REFERENCES `clinicaesteticasena`.`citas` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `consentimientos_ibfk_2`

FOREIGN KEY (`id_usuario`)


REFERENCES `clinicaesteticasena`.`usuarios` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB

```

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 28 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

AUTO\_INCREMENT = 2

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4\_0900\_ai\_ci;

-----

-- Table `clinicaesteticasena`.`examens`

-----

CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`examens` (

  `id` INT NOT NULL AUTO\_INCREMENT,

  `id\_cita` INT NULL DEFAULT NULL,

  `nombre\_examen` VARCHAR(255) NULL DEFAULT NULL,

  `archivo\_examen` VARCHAR(255) NULL DEFAULT NULL,

  `observaciones` TEXT NULL DEFAULT NULL,

  `createdAt` DATETIME NOT NULL,

  `updatedAt` DATETIME NOT NULL,


  PRIMARY KEY (`id`),

  INDEX `id\_cita` (`id\_cita` ASC) VISIBLE,

  CONSTRAINT `examens\_ibfk\_1`

    FOREIGN KEY (`id\_cita`)

      REFERENCES `clinicaesteticasena`.`citas` (`id`)

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 29 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB

AUTO\_INCREMENT = 3

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4\_0900\_ai\_ci;

-- -----

-- Table `clinicaesteticasena`.`historialclinicos`

-- -----

CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`historialclinicos` (

  `id` INT NOT NULL AUTO\_INCREMENT,

  `id\_usuario` INT NOT NULL,

  `enfermedades` TEXT NULL DEFAULT NULL,

  `alergias` TEXT NULL DEFAULT NULL,

  `cirugias\_previas` TEXT NULL DEFAULT NULL,


  `condiciones\_piel` TEXT NULL DEFAULT NULL,

  `embarazo\_lactancia` TINYINT(1) NULL DEFAULT NULL,

  `medicamentos` TEXT NULL DEFAULT NULL,

  `consume\_tabaco` TINYINT(1) NULL DEFAULT NULL,



	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página <b>30</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

```

`consume_alcohol` TINYINT(1) NULL DEFAULT NULL,

`usa_anticonceptivos` TINYINT(1) NULL DEFAULT NULL,

`detalles_anticonceptivos` TEXT NULL DEFAULT NULL,

`diabetes` TINYINT(1) NULL DEFAULT NULL,

`hipertension` TINYINT(1) NULL DEFAULT NULL,

`historial_cancer` TINYINT(1) NULL DEFAULT NULL,

`problemas_coagulacion` TINYINT(1) NULL DEFAULT NULL,

`epilepsia` TINYINT(1) NULL DEFAULT NULL,

`otras_condiciones` TEXT NULL DEFAULT NULL,

`createdAt` DATETIME NOT NULL,

`updatedAt` DATETIME NOT NULL,

PRIMARY KEY (`id`),

UNIQUE INDEX `id_usuario` (`id_usuario` ASC) VISIBLE,

CONSTRAINT `historialclinicos_ibfk_1`

FOREIGN KEY (`id_usuario`)

REFERENCES `clinicaesteticasena`.`usuarios` (`id`)

ON DELETE CASCADE


ON UPDATE CASCADE)

ENGINE = InnoDB

AUTO_INCREMENT = 3

DEFAULT CHARACTER SET = utf8mb4

```

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página <b>31</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

COLLATE = utf8mb4\_0900\_ai\_ci;

-- -----

-- Table `clinicaesteticasena`.`mensaje`

-- -----

CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`mensaje` (

  `id` INT NOT NULL AUTO\_INCREMENT,

  `id\_usuario` INT NOT NULL,

  `id\_asistente` INT NULL DEFAULT NULL,

  `id\_cita` INT NULL DEFAULT NULL,

  `remitante` ENUM('usuario', 'asistente') NOT NULL,

  `texto` TEXT NOT NULL,

  `fecha\_envio` DATETIME NULL DEFAULT NULL,

  `leido` TINYINT(1) NULL DEFAULT '0',

  `createdAt` DATETIME NOT NULL,


  `updatedAt` DATETIME NOT NULL,

PRIMARY KEY (`id`),

INDEX `id\_usuario` (`id\_usuario` ASC) VISIBLE,

INDEX `id\_asistente` (`id\_asistente` ASC) VISIBLE,

INDEX `id\_cita` (`id\_cita` ASC) VISIBLE,


	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			Página <b>32</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

```

CONSTRAINT `mensaje_ibfk_1`
  FOREIGN KEY (`id_usuario`)
  REFERENCES `clinicaesteticasena`.`usuarios` (`id`)
  ON DELETE CASCADE
  ON UPDATE CASCADE,
CONSTRAINT `mensaje_ibfk_2`
  FOREIGN KEY (`id_asistente`)
  REFERENCES `clinicaesteticasena`.`usuarios` (`id`)
  ON DELETE SET NULL
  ON UPDATE CASCADE,
CONSTRAINT `mensaje_ibfk_3`
  FOREIGN KEY (`id_cita`)
  REFERENCES `clinicaesteticasena`.`citas` (`id`)
  ON DELETE SET NULL
  ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```


-----

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página <b>33</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

-- Table `clinicaesteticasena`.`notificaciones`

-- -----

```
CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`notificaciones` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `id_usuario` INT NOT NULL,
  `id_cita` INT NULL DEFAULT NULL,
  `tipo` ENUM('cita', 'confirmacion_cita', 'exámenes', 'sistema', 'cita_reagendada',
```

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 34 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

```

'cancelada', 'eliminada') NOT NULL,

`mensaje` TEXT NULL DEFAULT NULL,

`leida` TINYINT(1) NULL DEFAULT NULL,

`archivada` TINYINT(1) NULL DEFAULT NULL,

`fecha` DATETIME NULL DEFAULT NULL,

`createdAt` DATETIME NOT NULL,

`updatedAt` DATETIME NOT NULL,

PRIMARY KEY (`id`),

INDEX `id_usuario` (`id_usuario` ASC) VISIBLE,

INDEX `id_cita` (`id_cita` ASC) VISIBLE,

CONSTRAINT `notificaciones_ibfk_1`

FOREIGN KEY (`id_usuario`)

REFERENCES `clinicaesteticasena`.`usuarios` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `notificaciones_ibfk_2`

FOREIGN KEY (`id_cita`)


REFERENCES `clinicaesteticasena`.`citas` (`id`)

ON DELETE SET NULL

ON UPDATE CASCADE)

ENGINE = InnoDB

```

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 35 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

AUTO\_INCREMENT = 29

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4\_0900\_ai\_ci;

-----

-- Table `clinicaesteticasena`.`ordenprocedimientos`

-----

CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`ordenprocedimientos` (

  `id` INT NOT NULL AUTO\_INCREMENT,

  `id\_orden` INT NULL DEFAULT NULL,

  `id\_procedimiento` INT NULL DEFAULT NULL,

  `createdAt` DATETIME NOT NULL,

  `updatedAt` DATETIME NOT NULL,

  PRIMARY KEY (`id`),

  INDEX `id\_orden` (`id\_orden` ASC) VISIBLE,


  INDEX `id\_procedimiento` (`id\_procedimiento` ASC) VISIBLE,

  CONSTRAINT `ordenprocedimientos\_ibfk\_1`

    FOREIGN KEY (`id\_orden`)

      REFERENCES `clinicaesteticasena`.`ordenes` (`id`)

  ON DELETE CASCADE

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 36 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

```

ON UPDATE CASCADE,

CONSTRAINT `ordenprocedimientos_ibfk_2`

FOREIGN KEY (`id_procedimiento`)

REFERENCES `clinicaesteticasena`.`procedimientos` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB

AUTO_INCREMENT = 4

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4_0900_ai_ci;

-- -----

-- Table `clinicaesteticasena`.`procedimientoimagenes`

-- -----

CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`procedimientoimagenes` (

  `id` INT NOT NULL AUTO_INCREMENT,

  `url` VARCHAR(255) NOT NULL,


  `alt` VARCHAR(255) NULL DEFAULT NULL,

  `orden` INT NULL DEFAULT NULL,

  `procedimientold` INT NOT NULL,

```



	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página <b>37</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

```

`createdAt` DATETIME NOT NULL,

`updatedAt` DATETIME NOT NULL,

PRIMARY KEY (`id`),

INDEX `procedimientold` (`procedimientold` ASC) VISIBLE,

CONSTRAINT `procedimientoimagenes_ibfk_1`

FOREIGN KEY (`procedimientold`)

REFERENCES `clinicaesteticasena`.`procedimientos` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB

AUTO_INCREMENT = 10

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4_0900_ai_ci;

```

```

-- -----

```

```

-- Table `clinicaesteticasena`.`requerimientos`

```

```

-- -----

```


```

CREATE TABLE IF NOT EXISTS `clinicaesteticasena`.`requerimientos` (

`id` INT NOT NULL AUTO_INCREMENT,

`id_cita` INT NOT NULL,


```

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página <b>38</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

```

`descripcion` TEXT NULL DEFAULT NULL,
`frecuencia` INT NULL DEFAULT NULL,
`repeticiones` INT NULL DEFAULT NULL,
`fecha_inicio` DATE NULL DEFAULT NULL,
`fecha_fin` DATETIME NULL DEFAULT NULL,
`estado` ENUM('pendiente', 'realizada') NULL DEFAULT 'pendiente',
`createdAt` DATETIME NOT NULL,
`updatedAt` DATETIME NOT NULL,
PRIMARY KEY (`id`),
INDEX `id_cita` (`id_cita` ASC) VISIBLE,
CONSTRAINT `requerimientos_ibfk_1`
FOREIGN KEY (`id_cita`)
REFERENCES `clinicaesteticasena`.`citas` (`id`)
ON DELETE CASCADE
ON UPDATE CASCADE)
ENGINE = InnoDB
AUTO_INCREMENT = 4
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 39 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

## 7.7 Índices y rendimiento

- Índices creados en campos de búsqueda frecuente (correo, numerodocumento, id\_usuario, id\_cita).
- Consultas optimizadas mediante Sequelize Query Filters.
- Implementación de paginación y ordenamiento en los endpoints del backend para optimizar tiempos de respuesta.

## 7.8 Migraciones y versionado

- Migraciones generadas con Sequelize CLI (sequelize migration:generate).
- Archivos de migración almacenados en /backend/migrations con nombres secuenciales (YYYYMMDDHHMM-create-usuarios.js).
- Versionamiento sincronizado con GitHub, evitando modificaciones directas al esquema productivo.

## 7.9 Backups y restauración


- Backup local: exportado con MySQL Workbench o mediante el comando:  

```
mysqldump -u root -p clinicaesteticasena > backup_clinestetica.sql
```
- Restauración:  

```
mysql -u root -p clinicaesteticasena < backup_clinestetica.sql
```
- Entorno en la nube: copias automáticas gestionadas desde Render.

## 7.10 Monitoreo y mantenimiento

- Seguimiento de consultas lentas mediante logs de MySQL.
- Control de integridad referencial automatizado con Sequelize.
- Limpieza de registros antiguos mediante tareas programadas (cron jobs).

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			Página <b>40</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

- Verificación periódica de backups y restauraciones de prueba.

### 7.11 Alternativa en la nube

Para entornos de producción, Clinestetica puede integrar su base de datos en la nube mediante:

- Railway o Render: despliegue gestionado de MySQL.
- Cloudinary: almacenamiento en la nube de imágenes y documentos PDF.

### Conclusión

La base de datos de Clinestetica fue diseñada bajo criterios de normalización, seguridad y escalabilidad, garantizando la trazabilidad de la información clínica y administrativa.

Su implementación con MySQL + Sequelize permite un control eficiente y adaptable tanto en entornos locales como en producción, cumpliendo con las buenas prácticas de desarrollo establecidas por el SENA Regional Cauca – CTPI.

## 8. Funcionalidades del software


El sistema Clinestetica cuenta con un conjunto de funcionalidades diseñadas para la gestión integral de una clínica estética, permitiendo la administración de usuarios, citas, procedimientos, consentimientos informados, historiales clínicos y notificaciones, tanto desde la aplicación web como móvil.

Estas funcionalidades fueron validadas mediante pruebas unitarias e integrales, las cuales se describen detalladamente en el documento de Pruebas Unitarias\_Clinestetica, adjunto a este Manual Técnico.

Nota:

El documento de Pruebas Unitarias contiene el detalle técnico de cada funcionalidad probada, incluyendo:

- el nombre del módulo,
- los casos de prueba,
- el resultado esperado,
- el resultado obtenido,

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 41 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

- y el estado final (aprobado/rechazado).

#### 8.1. Módulo de Autenticación

- Registro de nuevos usuarios con validación de datos obligatorios.
- Inicio de sesión seguro mediante JWT (JSON Web Token).
- Recuperación de contraseña vía correo electrónico.
- Asignación de roles (doctor, asistente, usuario).

#### 8.2. Módulo de Usuarios

- Creación, edición, activación y eliminación de usuarios.
- Validación de documentos, correos y contraseñas cifradas.
- Gestión de roles y permisos de acceso.

#### 8.3. Módulo de Procedimientos


- Registro de procedimientos estéticos con descripción, duración, precio y categoría.
- Carga de imágenes y recomendaciones previas.
- Clasificación por categorías (toxina botulínica, rellenos, limpieza facial, etc.).

#### 8.4. Módulo de Citas

- Agendamiento de citas médicas o estéticas.
- Asignación de doctores y usuarios.
- Actualización de estado (pendiente, confirmada, realizada, cancelada).
- Registro de observaciones, notas de evolución y medicamentos recetados.

#### 8.5. Módulo de Consentimiento Informado

- Generación automática de documentos PDF firmados digitalmente.
- Almacenamiento en la nube mediante Cloudinary.
- Validación de integridad con hash y almacenamiento seguro.

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 42 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

## 8.6. Módulo de Notificaciones

- Generación automática de notificaciones por acciones del sistema (creación, confirmación o cancelación de citas).
- Gestión de notificaciones leídas, archivadas e históricas.
- Integración con Redis para optimizar el tiempo de entrega.

## 8.7. Módulo de Historial Clínico

- Registro de antecedentes médicos, cirugías, alergias y condiciones de salud.
- Consulta y actualización por parte del doctor o asistente autorizado.

## 8.8. Módulo de Comunicación

- Envío y recepción de mensajes entre usuario y asistente.
- Control de lectura y seguimiento de conversaciones.

## 8.9. Panel de Administración


- Dashboard general con métricas (número de usuarios, citas, procedimientos)
- Filtros de búsqueda, estadísticas básicas y control de estados.
- Acceso a logs del sistema y auditoría básica.

## 8.10. Módulo Móvil (Flutter)

- Visualización de citas, procedimientos y notificaciones desde la app móvil.
- Inicio de sesión con JWT y almacenamiento local con SharedPreferences
- Navegación mediante BottomNavigationBar y rutas nombradas.

Documento de referencia:

Anexo 2 – Documento, Pruebas Unitarias Clinestetica.

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 43 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

Contiene los casos de prueba correspondientes a cada módulo funcional descrito en este apartado, con los resultados obtenidos durante la validación del sistema.

## 9. Funcionalidades del Software

El sistema **Clinestetica** implementa un conjunto de funcionalidades orientadas a la **gestión integral de una clínica estética**, optimizando los procesos administrativos, clínicos y de comunicación entre los diferentes roles del sistema. El software se compone de tres perfiles principales (**Doctor, Asistente y Usuario**) que interactúan de forma dinámica a través de un entorno web y móvil sincronizado con la base de datos MySQL y servicios REST.


### 9.1 Funcionalidades Generales

- Autenticación y seguridad: sistema de registro, inicio de sesión y cierre de sesión mediante JWT (JSON Web Token), con control de roles y validación de credenciales encriptadas.
- Gestión de notificaciones: visualización en tiempo real de novedades, nuevas citas, reprogramaciones y subida de exámenes clínicos.
- Panel de control personalizado: interfaz dinámica adaptada al rol del usuario, mostrando módulos y accesos específicos.
- Gestión de citas: creación, edición, cancelación y reagendamiento con validaciones de horarios y disponibilidad médica.
- Consentimiento informado digital: visualización, firma y almacenamiento del documento legal previo a cada procedimiento estético.
- Historial médico: registro estructurado de antecedentes clínicos, motivo de consulta, evaluación médica y exámenes solicitados.
- Carga de exámenes clínicos: módulo de carga de archivos en formato PDF o imagen, con notificación automática al doctor.
- Diseño responsivo: interfaz compatible con dispositivos móviles, tablets y pantallas de escritorio.

### 9.2 Funcionalidades por Rol

#### a. Doctor

- Visualiza y gestiona las citas agendadas por los pacientes y asistentes.
- Registra y actualiza servicios o procedimientos estéticos disponibles en la clínica.
- Consulta historiales médicos y resultados de exámenes subidos por los pacientes.
- Recibe notificaciones en tiempo real sobre nuevas citas o consentimientos firmados.

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 44 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

- Genera informes y documentos clínicos en formato PDF.

#### b. Asistente

- Gestiona el agendamiento de citas de evaluación y procedimientos.
- Reagenda citas según indicaciones médicas y disponibilidad del doctor.
- Crea, edita y elimina servicios y categorías de procedimientos.
- Visualiza y completa historiales médicos en coordinación con el personal médico.
- Supervisa el registro de pacientes y mantiene actualizada la información clínica.

#### c. Usuario (Paciente)

- Se registra en la plataforma y accede a su perfil personal.
- Explora los procedimientos disponibles, con descripción, precios y recomendaciones previas.
- Agenda citas de evaluación y procedimientos seleccionando doctor, fecha y hora disponibles.
- Firma el consentimiento informado digital antes del procedimiento.
- Sube los exámenes requeridos desde su equipo, notificando automáticamente al doctor.
- Consulta, reagenda o cancela sus citas con al menos 24 horas de anticipación.
- Recibe notificaciones sobre confirmaciones, reprogramaciones y observaciones médicas.


### 9.3 Funcionalidades Técnicas Complementarias

- Integración con Cloudinary: para el almacenamiento de imágenes y documentos clínicos.
- API REST en Node.js y Express: para la comunicación entre frontend (React) y backend.
- Gestión de estados y notificaciones con Redis (opcional): mejorando el rendimiento y la respuesta en tiempo real.
- Manejo de errores y validaciones: mensajes personalizados para entradas incorrectas, archivos no válidos o intentos de acceso no autorizados.
- Soporte multirrol y modularidad: el sistema adapta automáticamente su interfaz y permisos según el rol autenticado.

### 9.4 Funcionalidades Móviles (Aplicación Flutter)

La aplicación móvil Clinestetica fue desarrollada en Flutter como extensión del sistema web, permitiendo a los usuarios y doctores acceder a los servicios de la clínica desde cualquier dispositivo Android.



	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 45 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

Esta versión móvil consume directamente la API REST implementada en Node.js y Express, comparte la misma base de datos MySQL, y mantiene la autenticación y roles mediante JWT.

Su diseño sigue los lineamientos de la identidad institucional, priorizando la usabilidad, el contraste visual y la coherencia con la interfaz web.

#### 9.4.1 Funcionalidades del Usuario (Paciente)


El rol Usuario cuenta con una experiencia interactiva centrada en la autogestión de su proceso clínico y estético. Entre sus principales funcionalidades se encuentran:

- Visualización de servicios: el paciente puede explorar el catálogo de procedimientos estéticos (toxina botulínica, radiofrecuencia facial, drenaje linfático, etc.) con información detallada de duración, precio e imagen ilustrativa.
- Inicio de sesión y registro: acceso mediante correo y contraseña o autenticación social (Google / Facebook), con validación de campos y alertas de éxito.
- Selección de servicios y reserva: el usuario elige un servicio, lo añade al carrito, consulta la disponibilidad médica y agenda su cita según el horario del doctor.
- Panel de usuario: acceso directo a los módulos principales:
  - *Mis Citas* → visualización y control de citas agendadas.
  - *Mi Historial Médico* → revisión de antecedentes y registros previos.
  - *Notificaciones* → recepción de alertas en tiempo real.
  - *Mi Perfil* → edición y actualización de datos personales.
  - *Ayuda* → contacto y soporte en línea.
- Sección “Acerca de Clinestética”: incluye descripción institucional, servicios destacados, ubicación, número de contacto y enlace directo al sitio web oficial.
- Confirmaciones visuales: mensajes modales de éxito, como “*Login exitoso, bienvenido usuario*”, refuerzan la retroalimentación positiva.
- Diseño responsivo: adaptado a diferentes resoluciones y orientaciones de pantalla, implementando componentes Material Design.

#### 9.4.2 Funcionalidades del Doctor

El rol Doctor dispone de una versión móvil optimizada que le permite supervisar y gestionar de manera ágil su agenda y pacientes. Las principales características son:

- Panel Doctor: acceso principal a los módulos *Citas Hoy*, *Pacientes*, *Historial Clínico*, *Notificaciones* y *Perfil*, organizados en tarjetas de colores distintivos.
- Gestión de citas: visualización de las citas del día y del mes mediante un calendario interactivo; permite consultar fecha, hora, tipo y estado de cada cita.

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			<b>Página 46 de 50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

- Gestión de pacientes: listado completo de usuarios registrados con búsqueda por nombre, documento o correo electrónico; acceso a los detalles individuales (rol, contacto, estado, género).
- Historial clínico: acceso a los historiales de cada paciente, donde se visualizan datos clínicos relevantes (enfermedades, alergias, cirugías, condiciones de piel, medicamentos y antecedentes).
- Notificaciones: recepción de avisos sobre nuevas citas, carga de exámenes clínicos y actualizaciones de estado.
- Perfil del doctor: visualización y edición de información personal (nombre, correo, documento, teléfono, dirección, fecha de nacimiento, ocupación, estado civil y aceptación de términos).
- Diseño visual: interfaz coherente con el panel de usuario, colores institucionales (verde, blanco y acentos en violeta), y organización clara para optimizar la atención médica desde dispositivos móviles.


## 10. Interfaces del Sistema (UI/UX)

### 10.1 Lineamientos generales de diseño

La interfaz del sistema Clinestetica fue desarrollada con enfoque responsive, manteniendo coherencia visual entre los módulos web y móvil. El diseño utiliza una paleta cromática basada en tonos neutros (gris claro, blanco y negro) combinada con acentos en violeta y verde, que representan estética, salud y tecnología. La tipografía base es Inter / Roboto, priorizando la legibilidad y jerarquía visual. Se emplean íconos de Lucide y CoreUI para reforzar la claridad de las acciones, y botones con estados activos, deshabilitados y de confirmación. Los componentes se diseñaron siguiendo principios de usabilidad, accesibilidad y consistencia visual, manteniendo márgenes amplios y una experiencia fluida en escritorio y dispositivos móviles.

### 10.2 Arquitectura de navegación

La aplicación cuenta con un menú lateral fijo que centraliza el acceso a los módulos principales, dependiendo del rol del usuario autenticado. La barra superior muestra el rol activo, el menú de usuario y el sistema de notificaciones. El flujo de navegación se estructura de la siguiente manera:

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			Página <b>47</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

- Inicio / Login / Registro  
Acceso al sistema y creación de cuentas nuevas con validaciones de campos y mensajes de error contextualizados.
- Dashboard (según rol)  
Visualización de tarjetas interactivas para acceder a funciones clave.
- Módulos secundarios  
Navegación jerárquica desplegable para gestión de servicios, citas, usuarios, procedimientos y notificaciones.

### 10.3 Interfaces por Rol

#### a. Panel Doctor

El doctor cuenta con un Dashboard personalizado donde puede:

- Visualizar citas programadas y acceder a detalles clínicos.
- Registrar servicios o procedimientos nuevos (como toxina botulínica, radiofrecuencia, masajes, etc.).
- Acceder a historiales médicos, actualizar información de pacientes y generar informes PDF.
- Revisar notificaciones en tiempo real (registro de nuevas citas, exámenes subidos, reprogramaciones).


La interfaz mantiene un diseño profesional con fondo oscuro en los módulos principales, íconos destacados y secciones bien jerarquizadas para optimizar la atención médica.

#### b. Panel Asistente

El asistente cumple funciones administrativas y de apoyo clínico. Su panel incluye:

- Citas: consultar, reagendar y registrar nuevas citas para pacientes.
- Procedimientos: crear, editar, eliminar y clasificar servicios o categorías.
- Historiales: completar o revisar información médica ingresada por el usuario o el doctor.

La interfaz (ver imagen del *Panel Asistente*) presenta una organización en tarjetas funcionales (Citas, Agendar Cita, Procedimientos) con acceso rápido desde el menú lateral a opciones adicionales: “Registrar Cita”, “Reagendar Cita”, “Completar Historial”, “Crear Servicio”, “Consultar Categorías”. Los formularios mantienen validaciones dinámicas y mensajes de confirmación visual con colores diferenciados (verde para éxito, rojo para error).

	<b>Centro de Teleinformática y Producción Industrial - Regional Cauca</b>			Página <b>48</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

### c. Panel Usuario (Paciente)

El usuario dispone de una interfaz orientada a la autogestión de su proceso clínico:

- Explorar servicios con imágenes, precios y descripciones (radiofrecuencia, drenajes, masajes, etc.).
- Reservar citas seleccionando el servicio, añadiéndolo al carrito y confirmando fecha, hora y doctor.
- Llenar historial médico mediante un formulario estructurado con condiciones de salud, alergias y medicamentos.
- Firmar consentimiento informado antes de cada procedimiento, confirmando aceptación de riesgos y beneficios.
- Subir exámenes clínicos requeridos y notificar automáticamente al doctor.
- Consultar y reagendar citas con antelación mínima de 24 horas. El diseño prioriza la simplicidad, con tarjetas claras, tipografía de alto contraste y botones grandes para interacción táctil.

### 10.4 Accesibilidad y experiencia de usuario


- Cumplimiento parcial de pautas WCAG 2.1 nivel AA (contraste mínimo 4.5:1, navegación con teclado, textos alternativos en iconos).
- Componentes responsivos diseñados en React y Flutter (en versión móvil), adaptándose a diferentes tamaños de pantalla.
- Alertas y notificaciones visuales mediante toasts y badges numéricos.
- Validaciones con mensajes inmediatos (“onBlur”) para evitar errores de usuario.
- 

### 10.5 Estados y rendimiento

- Cargando: indicadores animados (spinners) cuando se procesan peticiones API.
  - Vacío: mensajes con guías (“Aún no tienes citas registradas”).
  - Error: alertas con descripción y botón de reintento.
  - Éxito: confirmaciones breves en color verde (“Consentimiento firmado correctamente”).
- El sistema mantiene un tiempo de respuesta promedio < 2s y una tasa de refresco de 60 fps en componentes dinámicos. Se emplea lazy loading para optimizar carga de imágenes y rutas protegidas.

### 10.6 Interfaces Móviles (UI/UX de la App Flutter)

La versión móvil de Clinestetica mantiene la línea estética de la plataforma web, con énfasis en accesibilidad, simplicidad y consistencia visual.

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página 49 de 50
	Número de Documento:	FS-DOC Formato Manual Técnico	Fecha de Creación: 01/02/2025	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	Formato para la construcción del Manual Técnico		

### 10.6.1 Lineamientos de diseño


- Paleta cromática: verde institucional, blanco y gris claro, transmitiendo bienestar, salud y profesionalismo.
- Tipografía: *Inter / Roboto*, priorizando legibilidad y jerarquía visual.
- Iconografía: sistema *Material Icons* y *Lucide* para reforzar claridad y acción.
- Componentes: uso de *Cards*, *AppBar*, *Drawer*, *BottomNavigationBar* y *Dialogs* con esquinas redondeadas y sombras suaves.

### 10.6.2 Pantallas principales

1. Login / Registro: formulario de autenticación con integración a redes sociales.
2. Servicios: listado de procedimientos disponibles con duración y precio.
3. Detalles del servicio: información ampliada del tratamiento seleccionado.
4. Carrito y reserva: resumen de servicios antes de agendar la cita.
5. Panel Usuario: acceso a *Mis Citas*, *Historial*, *Notificaciones*, *Perfil* y *Ayuda*.
6. Panel Doctor: tarjetas de acceso rápido a *Citas*, *Pacientes*, *Historial*, *Notificaciones* y *Perfil*.
7. Calendario de citas: visualización interactiva con eventos programados.
8. Detalle del paciente: datos personales, contacto y estado.
9. Historial clínico: ficha médica editable y estructurada por secciones.
10. Notificaciones: listado de eventos con íconos y fecha de actualización.
11. Perfil (doctor / usuario): visualización y edición de información básica.

### 10.6.3 Experiencia de usuario

- Accesibilidad: contraste  $\geq 4.5 : 1$ , navegación con teclado y etiquetas descriptivas.
- Rendimiento: tiempo de carga  $< 2$  s, animaciones a 60 fps y *lazy loading* de imágenes.
- Retroalimentación visual: estados de éxito, error y vacío, mensajes “toast” y *badges* numéricos.
- Coherencia multiplataforma: diseño equivalente en web y móvil para garantizar una experiencia uniforme.

	Centro de Teleinformática y Producción Industrial - Regional Cauca			Página <b>50</b> de <b>50</b>
	Número de Documento:	<b>FS-DOC Formato Manual Técnico</b>	Fecha de Creación: <b>01/02/2025</b>	Elaborado por: Grupo instructores área de software
	Nombre del Documento:	<b>Formato para la construcción del Manual Técnico</b>		

## Bibliografía

- Beck, K. (2003). Test-driven development: By example. Addison-Wesley.
- CoreUI. (s. f.). React UI Components Library. Recuperado el 20 de septiembre de 2025, de <https://coreui.io/react/>
- Docker. (2025). Get Started with Docker. Docker Documentation. Recuperado de <https://docs.docker.com/get-started/>
- Node.js Foundation. (2025). Node.js v18.x Documentation. Recuperado de <https://nodejs.org/en/docs/>
- Sequelize. (2025). Sequelize: Node.js ORM for SQL Databases. Recuperado de <https://sequelize.org/>