

Documento de Casos de Uso (Texto Descriptivo)

1. Definición y Alcance

Los **Casos de Uso** describen la interacción del actor con el sistema para lograr un objetivo específico. Cada caso se detalla en sus pasos, precondiciones, y referencias a la arquitectura del proyecto.

1.1. Actores del Sistema

El sistema cuenta con cuatro actores principales, definidos por su rol en el *backend* (backend/src/models/User.js):

Actor	Rol	Descripción	Acceso Principal
Paciente	paciente	Consulta disponibilidad, reserva, paga y cancela citas.	patient_dashboard.html
Médico	medico	Visualiza agenda, cierra citas y gestiona órdenes médicas.	doctor_dashboard.html
Admin de Centro	admin_centro	Gestiona al personal y los recursos de su centro asignado.	medical_center_dashboard.html
Admin del Sistema	admin_sistema	Mantenimiento global del sistema y acceso a métricas.	administrator_dashboard.html

2. Referencia Rápida: Listado Completo de Casos de Uso

Este listado incluye todos los casos de uso cubiertos por los Requisitos Funcionales (RF) del sistema.

2.1. Casos de Uso del Paciente (Flujos de Consumo)

CU ID	Objetivo Principal	RF ID Asociado
CU-01	Iniciar Sesión y acceder al sistema.	RF-02, RF-04
CU-05	Buscar Disponibilidad de citas y profesionales (RF-06).	RF-06, RF-07
CU-06	Reservar Nueva Cita (Creación).pdf].	RF-07, RF-08
CU-07	Gestionar Citas (Ver listado, Cancelar cita).pdf].	RF-09, RF-10
CU-08	Realizar Pago de citas (online, con descuentos).	RF-22, RF-23
CU-09	Consultar Puntos y Recompensas.	RF-24
CU-10	Enviar Mensaje a Médico (Chat).	RF-20

2.2. Casos de Uso del Médico (Flujos de Servicio)

CU ID	Objetivo Principal	RF ID Asociado
CU-11	Gestionar Agenda (Ver listado de citas, Marcar como completada).	RF-10, RF-11
CU-12	Generar Orden Médica.	RF-19
CU-13	Responder Mensaje a Paciente (Chat).	RF-20
CU-14	Consultar Perfil y Disponibilidad.	RF-03

2.3. Casos de Uso del Administrador (Flujos de Mantenimiento)

CU ID	Objetivo Principal	RF ID Asociado
CU-15	Gestionar Usuarios (CRUD completo y roles).	RF-12, RF-13, RF-14, RF-15, RF-18
CU-16	Gestionar Centros Médicos (CRUD y control de estado).	RF-16, RF-17
CU-17	Consultar Métricas y Analíticas del sistema.	RF-25
CU-18	Consultar Notificaciones y Logs de Actividad.	RF-21

3. Casos de Uso Descriptivos Clave (Detalle del Flujo)

3.1. CU-01: Iniciar Sesión en la Plataforma

Atributo	Descripción
ID	CU-01 (RF-02)
Nombre	Iniciar Sesión
Actor Principal	Todos
Prioridad	Crítica (P0)
Objetivo	Autenticar al usuario para acceder a su panel.
Precondición	El actor debe estar previamente registrado en la colección User.
Postcondición	El sistema otorga un token JWT válido y redirige al <i>Dashboard</i> correspondiente (RF-04).
Flujo Normal	<p>1. El actor accede a /login.</p> <p>2. El <i>frontend</i> (web/medical_appointment_login_page.html) llama a authAPI.login(email, password) en web/js/api.js.</p> <p>3. El <i>backend</i> recibe POST /api/auth/login (backend/src/routes/auth.routes.js) y verifica credenciales (Bcrypt).</p> <p>4. El <i>backend</i> genera el token y lo devuelve.</p> <p>5. El <i>frontend</i> guarda el token y el rol en <i>localStorage</i> y redirige (redirectToDashboard) al <i>Dashboard</i> (RF-03).</p>
Flujos Alternativos	<p>A. Credenciales Inválidas: El <i>backend</i> devuelve un error 401. El <i>frontend</i> muestra un mensaje de error.</p> <p>B. Acceso Directo No Autorizado: Si el actor intenta acceder a una ruta protegida sin sesión, la función protectPage() en web/js/api.js lo redirige inmediatamente a /login.</p>

3.2. CU-06: Reservar Nueva Cita (Flujo Principal)

Atributo	Descripción
ID	CU-06 (RF-08)
Nombre	Reserva de Cita Médica
Actor Principal	Paciente
Prioridad	Alta (P1)
Objetivo	Crear un registro de cita médica, con pago y confirmación.
Precondición	El Paciente está autenticado.
Postcondición	Se crea un nuevo registro en el modelo Appointment.js con estado "Pendiente" y se realiza el cargo económico.
Flujo Normal	<p>1. El Paciente accede a "Nueva Cita" (web/book_new_appointment.html).</p> <p>2. El Paciente selecciona Especialidad y Médico disponible (RF-06, RF-07).</p> <p>3. El Paciente selecciona una Fecha y Hora.</p> <p>4. El Paciente es redirigido a la pantalla de pago (web/online_payment_screen.html).</p> <p>5. El frontend calcula el total (RF-23).</p> <p>6. Tras el pago exitoso, se envía la petición de creación POST /api/appointments (RF-07).</p> <p>7. El sistema muestra un mensaje de éxito y redirige al Dashboard.</p>
Flujos Alternativos	<p>A. Cancelación: El Paciente puede cancelar la cita desde web/patient_dashboard.html llamando a DELETE /api/appointments/:id (RF-10).</p> <p>B. Pago Rechazado: La interfaz de pagos notifica el error (ej., tarjeta inválida) y no se envía la petición de creación de cita.</p>

3.3. CU-15: Mantenimiento y Gestión de Usuarios (CRUD)

Atributo	Descripción
ID	CU-15 (RF-12, RF-13, RF-14, RF-15)
Nombre	Gestión de Usuarios (CRUD)
Actor Principal	Admin del Sistema
Prioridad	Media (P2)
Objetivo	Permitir al <i>Admin del Sistema</i> mantener el directorio de usuarios y sus roles.
Precondición	El <i>Admin</i> está autenticado con rol admin_sistema (RNF-S3).
Postcondición	La colección <i>Users</i> se actualiza (creación, modificación, eliminación).
Flujo Normal	<p>1. El <i>Administrador</i> accede a web/administrator_dashboard.html.</p> <p>2. El <i>frontend</i> llama a usersAPI.list() (GET /api/users) para listar a todos los usuarios (RF-12).</p> <p>3. El <i>Admin</i> selecciona "Añadir Nuevo Usuario" o "Editar".</p> <p>4. El <i>frontend</i> abre un <i>modal</i> y el <i>Admin</i> completa los datos.</p> <p>5. Al enviar, se llama a usersAPI.create (POST /api/users) o usersAPI.update (PUT /api/users/:id) (RF-13, RF-14).</p> <p>6. El <i>backend</i> verifica el rol (checkRole) y realiza la operación.</p> <p>7. El <i>frontend</i> recarga la tabla dinámicamente (RF-18) y notifica el éxito.</p>
Flujos Alternativos	<p>A. Eliminación: El <i>Admin</i> puede seleccionar la opción "Eliminar" (DELETE /api/users/:id), previa confirmación.</p> <p>B. Permiso Denegado: Si un <i>Admin de Centro</i> sin permisos globales intenta modificar un usuario que no le corresponde, el <i>backend</i> devuelve un error 403 (Prohibido).</p>