**Entrega final**

*“Sistema de anulación de horas medicas”*

**Integrantes:**

André Espinoza.

Javier González.

Sebastián Hidalgo.

Matías Rietta.

Diego Robles.

Matías Sepúlveda.

**Docente:** Dr. Thierry de Saint Pierre.

Índice

[Empresa 3](#_Toc183370450)

[Contexto 3](#_Toc183370451)

[Desafío / Oportunidad 4](#_Toc183370452)

[Requerimientos 5](#_Toc183370453)

[Solución propuesta 11](#_Toc183370454)

[Metodología 17](#_Toc183370455)

# Empresa

RedSalud es una de las principales redes de atención médica en Chile, dedicada a ofrecer servicios de salud accesibles y de alta calidad. Con una amplia presencia en todo el país, RedSalud cuenta con una red de clínicas, centros médicos, y unidades de diagnóstico que buscan cubrir las necesidades de salud de sus pacientes de manera integral.

La compañía se enfoca en brindar una atención médica eficiente, segura, y personalizada, destacando su compromiso con la innovación en salud y la mejora continua de sus procesos. Además, RedSalud colabora con profesionales altamente capacitados y utiliza tecnología avanzada para asegurar la mejor atención a sus pacientes, abarcando diversas especialidades médicas y servicios ambulatorios.

Su objetivo principal es mejorar la calidad de vida de la población, proporcionando un acceso más inclusivo a servicios médicos confiables y de calidad.

# Contexto

RedSalud, un proveedor de servicios de salud enfrenta un desafío en la gestión de las cancelaciones de citas médicas por parte de los doctores. Actualmente, los médicos cancelan sus citas de forma manual mediante el envío de correos electrónicos, lo que ralentiza el proceso y provoca retrasos en la reprogramación. La solución propuesta es desarrollar una página web que automatice y agilice este proceso.

El objetivo es desarrollar una aplicación web intuitiva que permita a los médicos gestionar y visualizar sus citas médicas, así como cancelar aquellas que consideren necesarias. Además, el área administrativa podrá generar reportes y visualizar la información sobre las cancelaciones en un rango de tiempo definido por ellos. La aplicación incluirá un sistema de notificaciones automáticas para alertar al personal administrativo sobre nuevas solicitudes de suspensión o anulación de citas.

Asimismo, se proporcionará una herramienta que facilite la reprogramación de citas canceladas, con el fin de minimizar la pérdida de horas médicas y mejorar la experiencia del paciente.

# Desafío / Oportunidad

Red Salud, un proveedor de servicios de salud reconocido, enfrenta actualmente una serie de contratiempos que impactan negativamente la calidad de sus servicios. Entre los principales problemas se encuentran la alta demanda de atención médica, la falta de disponibilidad constante de médicos y la frecuente cancelación o suspensión imprevista de citas. Estos factores generan retrasos en la atención, afectan la eficiencia operativa y, en última instancia, disminuyen la satisfacción del cliente, afectando la reputación de la organización en un entorno donde la calidad del servicio es crítica.

Para abordar estos desafíos, nuestro equipo de estudiantes de Ingeniería en Informática ha propuesto una solución tecnológica innovadora que optimiza la gestión de citas médicas. El objetivo es minimizar la pérdida de citas debido a cancelaciones y suspensiones mediante una aplicación específica para la gestión de citas. Esta herramienta permitirá mejorar la eficiencia en la administración de las agendas de los profesionales de salud y facilitará la reprogramación rápida de citas canceladas, garantizando así una experiencia fluida tanto para los médicos como para los pacientes.

Este proyecto es una oportunidad clave para que los estudiantes de Ingeniería en Informática apliquen sus conocimientos en un contexto real y desafiante. La importancia de este desafío se centra en:

* **Aplicación Práctica**: Permite a los estudiantes integrar habilidades en programación, bases de datos y diseño de interfaces, demostrando su capacidad para resolver un problema concreto en el sector salud.
* **Desarrollo de Competencias Profesionales**: Gestionar este proyecto implica aplicar metodologías de gestión, asignar tareas, seguir el progreso y comunicarse con stakeholders, simulando un entorno profesional.
* **Innovación en el Sector Salud**: Participar en la mejora de la eficiencia en la gestión de citas médicas aporta experiencia en un área de alto impacto social y en constante crecimiento.
* **Responsabilidad Social**: Contribuir a la optimización del servicio en salud no solo tiene un impacto académico, sino que también ayuda a mejorar la experiencia de los pacientes, mostrando cómo la tecnología puede beneficiar a la sociedad.

Este proyecto, más allá de un reto académico, es una oportunidad para que los estudiantes demuestren su capacidad de innovación y su compromiso con la mejora de servicios esenciales.

Stakeholders

* Carmen O'Kuinghttons
* Dalila Silva

Equipo de trabajo y roles

El equipo de trabajo para el desarrollo de este proyecto está compuesto por los siguientes miembros y sus respectivas funciones son:

* *André Espinoza:* Scrum master, developer, encargado de la redacción de informes y jefe de control.
* *Javier González:* developer, desarrollo de diagramas y encargado de la base de datos
* *Sebastián Hidalgo:* developer y encargado del desarrollo de las presentaciones.
* *Matías Rietta:* developer frontend y backend.
* *Diego Robles:* developer, desarrollo de diagramas y encargado de la base de datos.
* *Matías Sepúlveda:* developer, desarrollo de diagramas.

# Requerimientos

Captura del Requerimiento

1. **Criterios de Aceptación de Cancelación de Hora**.
2. **Habilitar o Deshabilitar la Aceptación Automática** (para administrativos).
3. **Autenticación**: El sistema debe permitir el ingreso con RUT y contraseña.
4. **Rechazo Automático**: El sistema debe rechazar solicitudes automáticamente si la fecha es muy cercana.
5. **Gestión de Horas Canceladas**: El sitio web debe contar con un sistema para gestionar las horas canceladas.
6. **Independencia del Sistema**: El sistema debe ser independiente y no integrado con otros sistemas.
7. **Escalabilidad**: El sistema debe ser escalable para adaptarse a un aumento en el uso.
8. **Ajuste Visual**: El sistema debe asemejarse lo más posible al estilo gráfico de RedSalud.

Análisis y Refinamiento de Requerimientos

Requerimientos Funcionales (RF)

1. **RF-01: Registro de Médico** - El sistema debe registrar nuevos médicos.
2. **RF-02: Eliminar Médico** - El sistema debe permitir eliminar un médico.
3. **RF-03: Editar Médico** - El sistema debe permitir editar los datos de un médico.
4. **RF-04: Ingreso de Login** - El sistema debe permitir el ingreso al sistema.
5. **RF-05: Registrar Administrador** - El sistema debe registrar un administrador.
6. **RF-06: Editar Administrador** - El sistema debe permitir editar los datos de un administrador.
7. **RF-07: Eliminar Administrador** - El sistema debe permitir eliminar un administrador.
8. **RF-08: Registrar jefe de Turno** - El sistema debe registrar un usuario como jefe de turno.
9. **RF-09: Editar jefe de Turno** - El sistema debe permitir editar los datos de un jefe de turno.
10. **RF-10: Ingreso de Formulario** - El sistema debe permitir el ingreso a un formulario con datos específicos.
11. **RF-11: Registro de Formulario** - El sistema debe redirigir a un formulario específico.
12. **RF-12: Confirmar Solicitud** - El sistema debe confirmar la solicitud ingresada.
13. **RF-13: Confirmar Solicitud con Prioridad** - El sistema debe rechazar automáticamente solicitudes con fechas cercanas.
14. **RF-14: Validar Contraseña de Médico** - El sistema debe validar la contraseña de un médico.
15. **RF-15: Validar Contraseña de Administrador** - El sistema debe validar la contraseña de un administrador.
16. **RF-16: Generar Notificación de Suspensión** - El sistema debe enviar notificaciones de suspensión.
17. **RF-17: Generar Informe** - El sistema debe generar informes con los datos.
18. **RF-18: Generar Métricas** - El sistema debe generar métricas con la información almacenada.
19. **RF-19: Verificación de Usuario** - El sistema debe verificar el tipo de usuario.
20. **RF-20: Actualización de Datos en el Dashboard** - El sistema debe actualizar los datos para la generación del dashboard.

Requerimientos No Funcionales (RNF)

1. **RNF-001: Mantenibilidad** - El sistema debe ser fácil de mantener y actualizar para reducir costos.
2. **RNF-002: Rendimiento** - Mantener tiempos de respuesta rápidos en momentos de alta demanda.
3. **RNF-003: Auditoría** - Registrar todas las acciones para garantizar la transparencia.
4. **RNF-004: Adaptabilidad** - El sistema debe ser flexible para cambios futuros.
5. **RNF-005: Confiabilidad** - Garantizar disponibilidad 24/7.
6. **RNF-006: Ajuste Gráfico RedSalud** - El sistema debe mantener los estándares gráficos de RedSalud.
7. **RNF-007: Modo Oscuro** - El sistema debe ofrecer un modo oscuro para reducir la fatiga visual.

Validación de Requerimientos

Los requerimientos han sido validados por la empresa y diseñados para cumplir con los objetivos del proyecto, minimizando la ambigüedad. Se han alineado con los procesos internos de la organización y han sido aprobados por los stakeholders clave, garantizando que cumplan con las necesidades de los usuarios y mejoren la eficiencia operativa.

Gestión de Requerimientos

Propósito

El proyecto tiene como objetivo optimizar la gestión de cancelaciones médicas. Se busca mejorar la eficiencia del sistema a través de funcionalidades clave como la autenticación, gestión de usuarios, cancelación de citas, visualización de métricas, y notificaciones automáticas.

Ámbito del Proyecto

El sistema operará de manera independiente, no integrado con la plataforma actual de RedSalud. Incluirá un sistema de inicio de sesión con distintos tipos de usuarios, cada uno con roles específicos.

Definiciones, Acrónimos y Abreviaturas

* **RF**: Requerimientos Funcionales.
* **RNF**: Requerimientos No Funcionales.

Referencias

* **ISO/IEC/IEEE 29148:2011** para levantamiento de requerimientos.

Descripciones Generales

Perspectiva del Producto

El sistema automatizado de gestión de cancelaciones optimizará la eficiencia al eliminar la dependencia de procesos manuales. Permitirá a los médicos cancelar citas directamente, reduciendo tiempos de espera y mejorando la asignación de recursos.

Funciones del Producto

* **Autenticación de Usuario**: Ingreso mediante RUT y contraseña.
* **Gestión de Usuarios**: Registro, edición, y eliminación de médicos, administradores y jefes de turno.
* **Formulario de Cancelación de Citas**: Permite cancelaciones con detalles específicos.
* **Rechazo Automático**: Cancelaciones rechazadas automáticamente según reglas.
* **Sistema de Aceptación Automática**: Configurable por administradores.
* **Dashboard Administrativo**: Visualización de métricas y gestión de horas suspendidas.
* **Notificaciones**: Alertas automáticas para cancelaciones y reprogramaciones.

Características de los Usuarios

* **Médicos**: Profesionales que atienden a los pacientes.
* **Cuerpo de Informática**: Encargados de tecnología e infraestructura.
* **Supervisor/jefe de Turno**: Responsable de operaciones y recursos.

Restricciones Generales

* **Escalabilidad**: Capaz de manejar un número creciente de usuarios sin perder rendimiento.
* **Confiabilidad**: Disponible las 24 horas, los 7 días de la semana.
* **Ajuste Visual**: Cumplir con los estándares gráficos de RedSalud.

Suposiciones y Dependencias

Inicialmente, se planteó como un procedimiento para que los médicos registren cancelaciones en una base de datos. Los administrativos podrían utilizar un dashboard para analizar estas cancelaciones y tomar decisiones en consecuencia.

Diagramas

Diagrama de proceso

Diagrama

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente

Diagrama de casos de uso

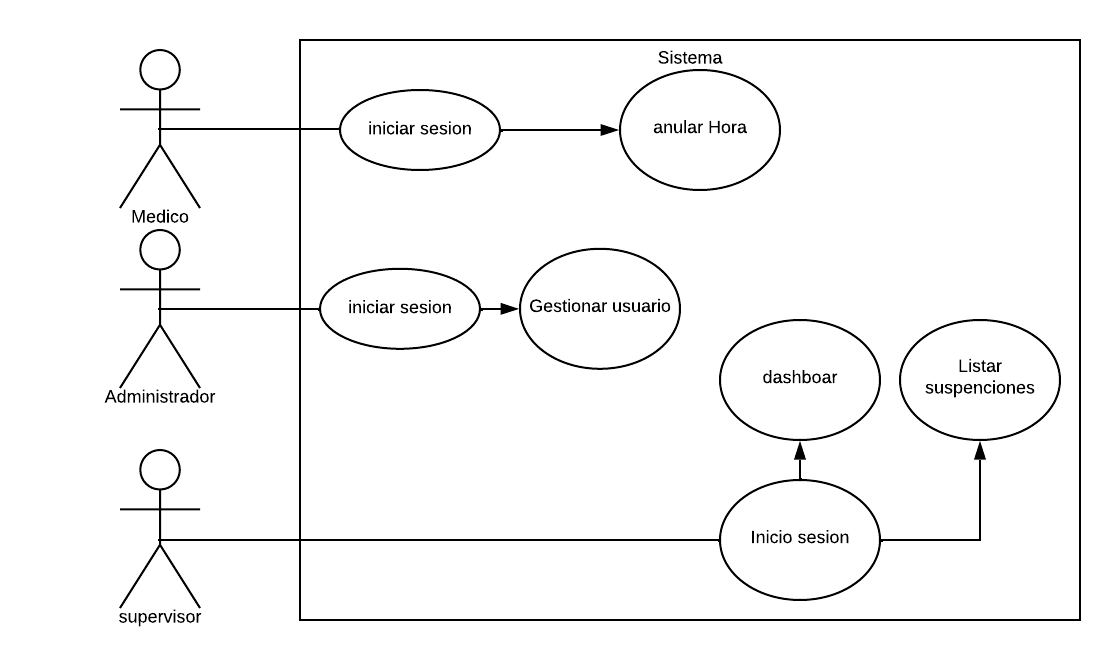


Diagrama de Secuencia del login

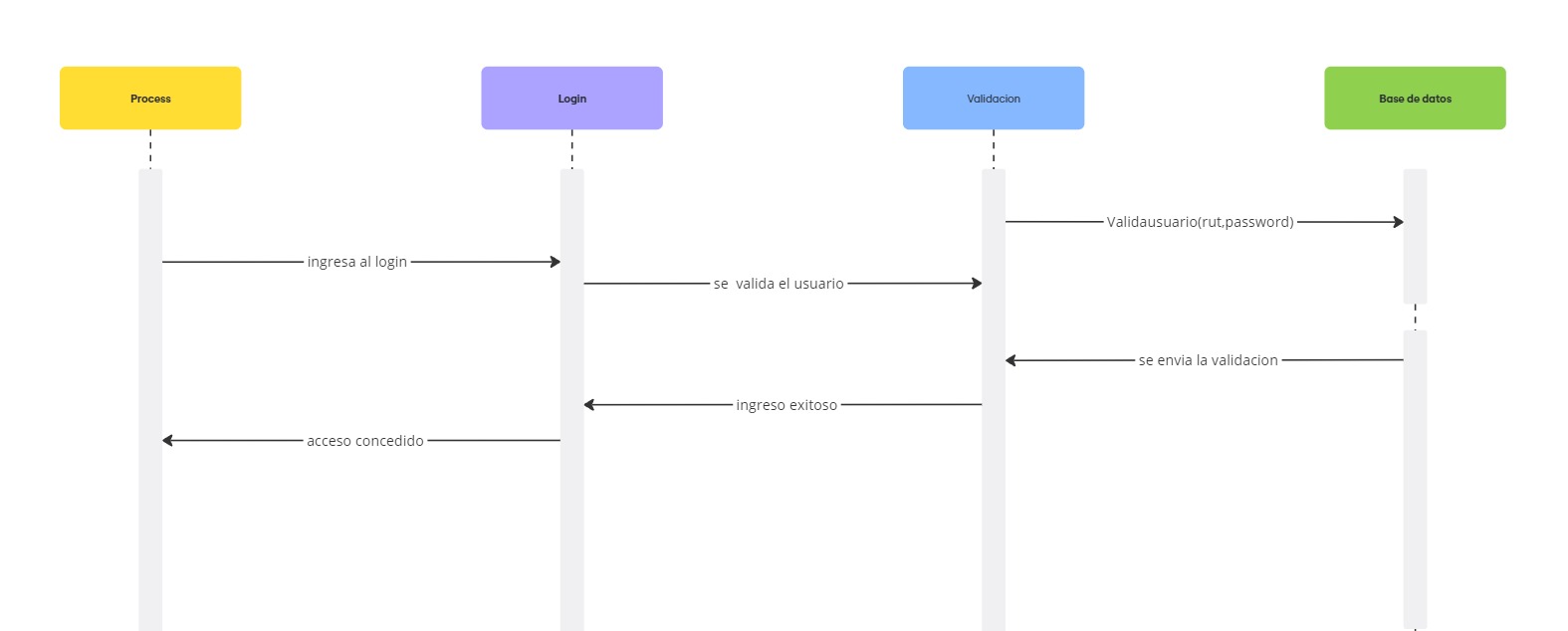


Diagrama de Secuencia formulario suspensión

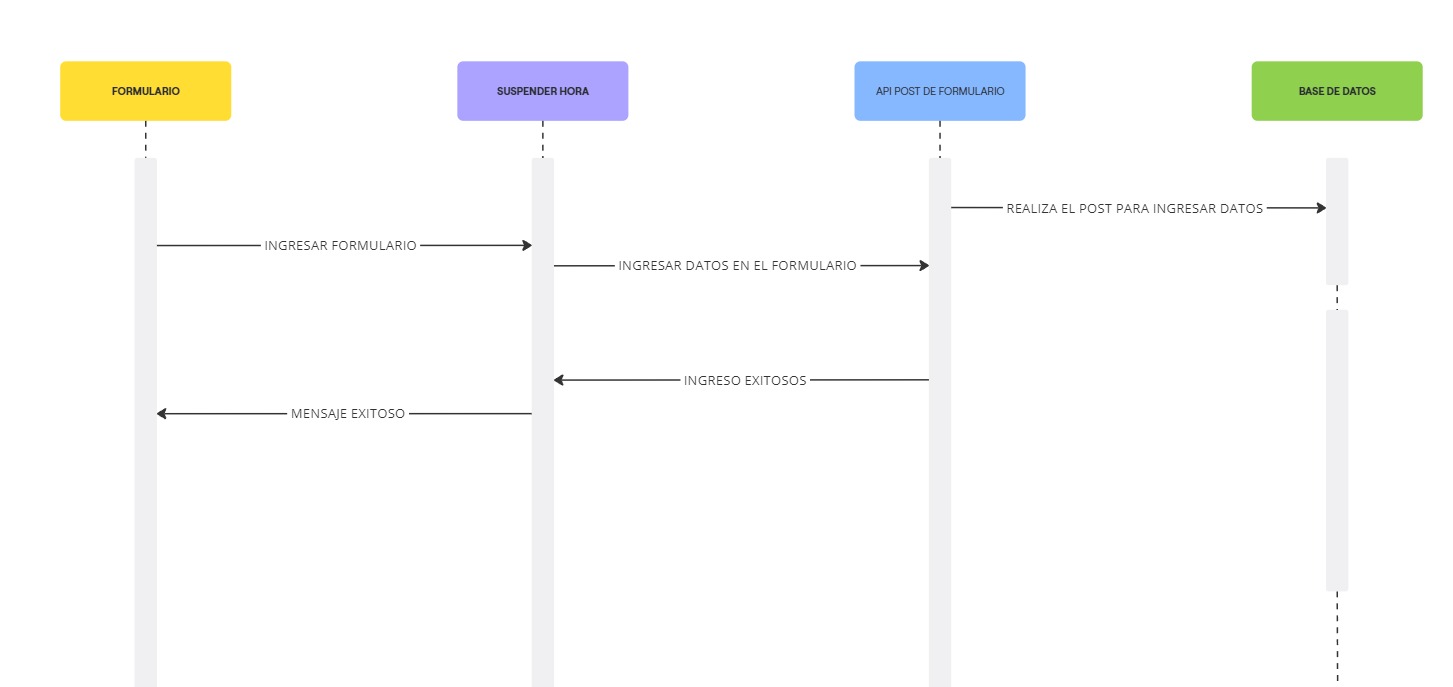


Diagrama de secuencia dashboard

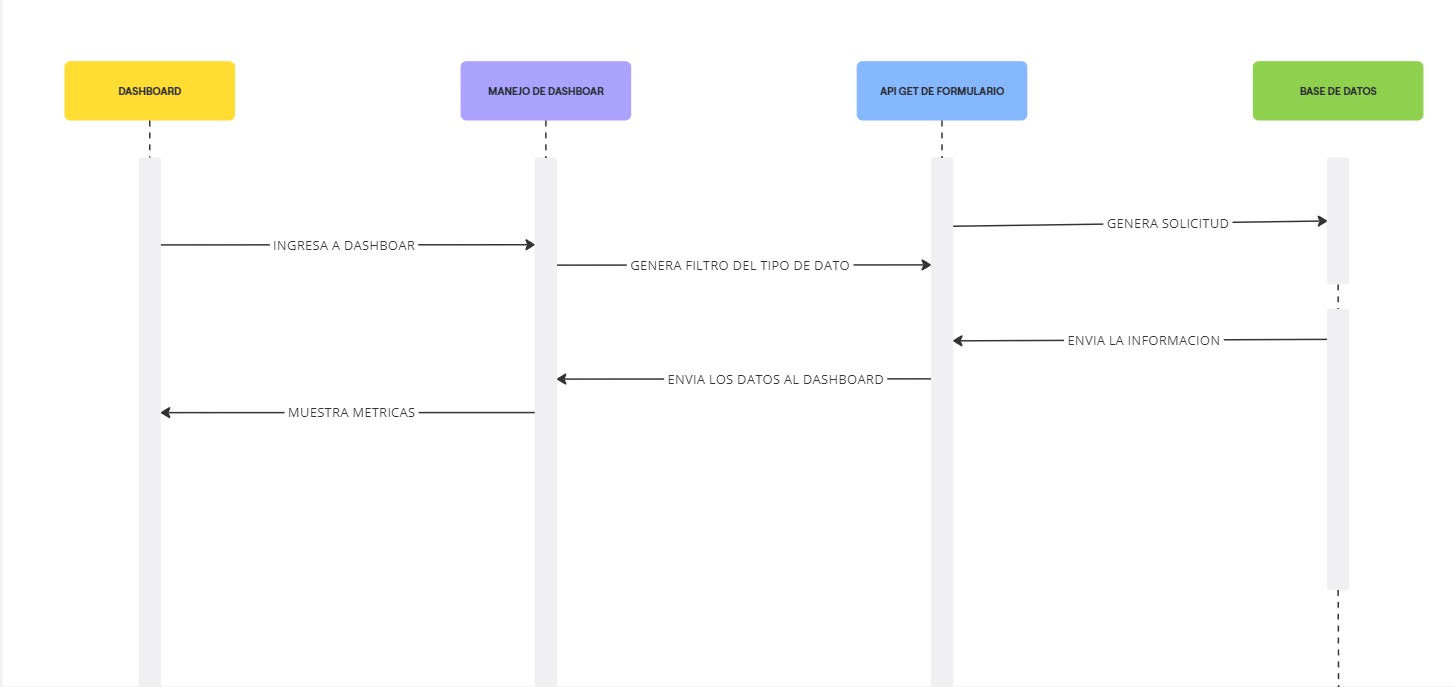
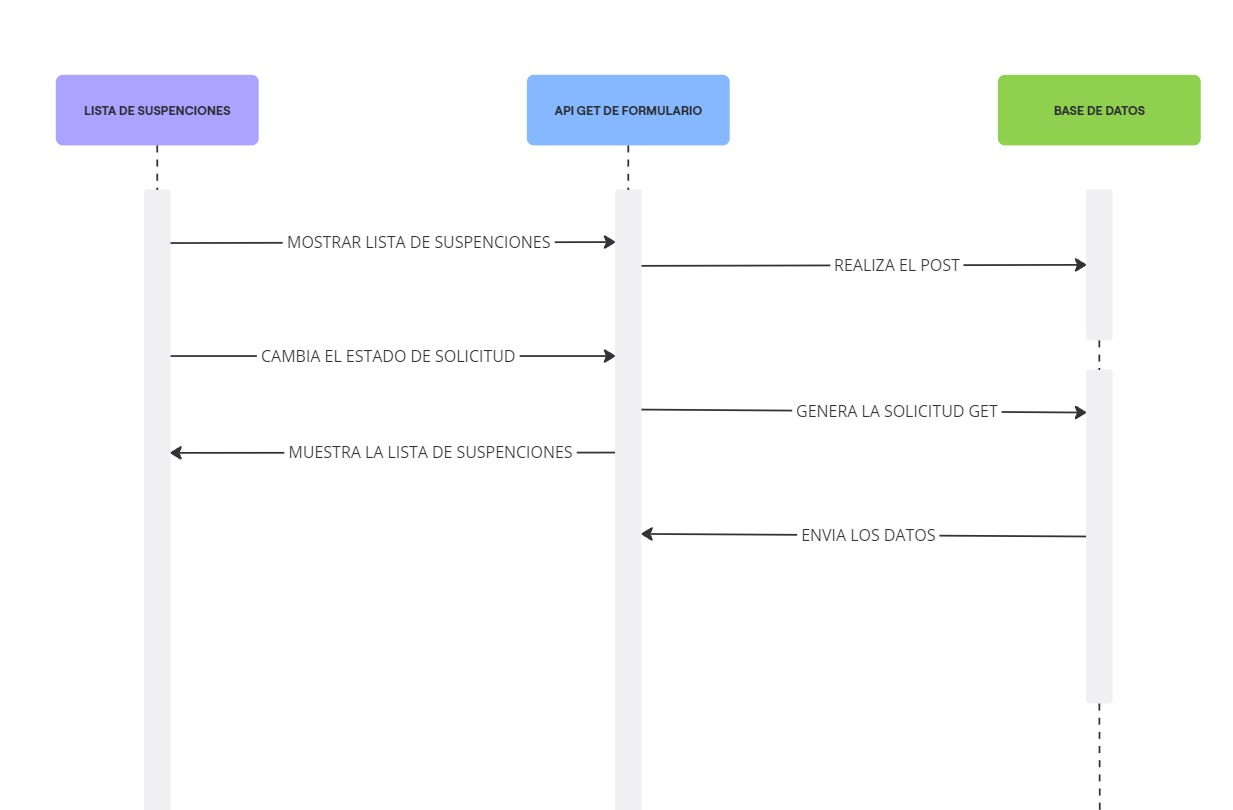


Diagrama de secuencia listar suspensiones



## Solución propuesta

a **solución propuesta** para el proyecto consiste en crear una **aplicación web** que facilite la gestión de cancelaciones de citas médicas para el personal médico de RedSalud, con las siguientes características clave:

**Objetivos y Funcionalidades Principales**

1. **Automatización del Proceso de Cancelación**:
   * Los médicos podrán cancelar citas directamente en el sistema sin necesidad de enviar correos electrónicos.
   * El sistema permitirá especificar detalles como la fecha, hora y motivo de la cancelación.
   * Existirá una función de **rechazo automático** para cancelaciones hechas con poca antelación, con reglas predefinidas.
2. **Notificaciones Automáticas**:
   * Notificaciones instantáneas tanto para pacientes como para médicos sobre cancelaciones y reprogramaciones.
   * Notificaciones al personal administrativo sobre nuevas solicitudes de cancelación.
3. **Sistema de Aceptación Automática**:
   * Los administradores podrán activar o desactivar la aceptación automática de solicitudes de cancelación.
4. **Dashboard Administrativo**:
   * El sistema incluirá un dashboard que permitirá a la administración visualizar métricas y estadísticas relacionadas con las cancelaciones.
   * Este dashboard facilitará la toma de decisiones mediante la visualización de informes detallados sobre cancelaciones y reprogramaciones.
5. **Reprogramación de Citas**:
   * El sistema ofrecerá una herramienta para la **reprogramación eficiente** de citas canceladas, minimizando la pérdida de horas médicas y mejorando la experiencia del paciente.

**Usuarios y Gestión de Roles**

1. **Autenticación de Usuarios**:
   * Ingreso seguro al sistema mediante RUT (Rol Único Tributario) y contraseña.
2. **Gestión de Usuarios**:
   * Registro, edición y eliminación de médicos, administradores y jefes de turno.
   * Validación de contraseñas según el rol del usuario (médico, administrador, jefe de turno).
3. **Verificación del Tipo de Usuario**:
   * El sistema determinará y gestionará las acciones permitidas según el rol del usuario.

**Requerimientos Técnicos**

1. **Criterios de Aceptación**:
   * Cancelaciones automáticas rechazadas si la fecha es demasiado cercana.
   * Independencia del sistema, es decir, será un sistema autónomo que no se integrará directamente con otras plataformas existentes de RedSalud.
   * Escalabilidad para soportar un número creciente de usuarios y operaciones sin comprometer el rendimiento.
2. **Requerimientos No Funcionales**:
   * **Confiabilidad**: El sistema debe estar disponible las 24 horas del día, los 7 días de la semana.
   * **Mantenibilidad**: Fácil mantenimiento y actualización.
   * **Auditoría**: Registro detallado de todas las acciones realizadas en el sistema para garantizar la transparencia.
   * **Adaptabilidad**: Flexibilidad para incorporar cambios futuros.
   * **Rendimiento**: Respuesta rápida incluso en momentos de alta demanda.

Para el diseño y arquitectura de la aplicación web enfocada en la gestión de cancelaciones de citas médicas para RedSalud, se propuso una solución estructurada que cumpla con los requisitos funcionales y no funcionales definidos. Aquí te presento una descripción detallada del diseño y la arquitectura, incluyendo los componentes principales y sus interacciones:

**1. Arquitectura del Sistema**

La arquitectura propuesta sigue un enfoque **MVC (Modelo-Vista-Controlador)**, utilizando tecnologías modernas para la construcción de aplicaciones web. El sistema se divide en tres capas principales:

**a. Capa de Presentación (Frontend)**

* **Tecnologías**: React (JavaScript), HTML, CSS.
* **Descripción**: Es la interfaz gráfica de la aplicación, donde los usuarios interactúan con el sistema. Se diseñarán componentes reutilizables que permitan mostrar la información de manera clara y ordenada.
* **Componentes Clave**:
  + **Login**: Página de autenticación para acceder al sistema.
  + **Dashboard Administrativo**: Panel para los usuarios administrativos con visualización de métricas y gestión de cancelaciones.
  + **Formulario de Cancelación**: Formulario accesible para médicos donde pueden cancelar citas.
  + **Gestión de Usuarios**: Formularios para la creación, edición y eliminación de usuarios (médicos, administradores, jefes de turno).
  + **Notificaciones**: Componentes para la visualización de notificaciones sobre cancelaciones y reprogramaciones.

**b. Capa de Lógica de Negocio (Backend)**

* **Tecnologías**: Django + Django REST Framework (Python).
* **Descripción**: Es la capa que maneja la lógica del negocio y las reglas que deben cumplirse para la gestión de cancelaciones y usuarios. Utiliza Django REST Framework para la creación de APIs RESTful.
* **Componentes Clave**:
  + **Controlador de Autenticación**: Maneja el acceso seguro al sistema usando RUT y contraseña.
  + **Gestor de Cancelaciones**: Controlador para la lógica de cancelación, incluyendo la validación automática de fechas y notificaciones.
  + **Gestor de Usuarios**: Controlador para el CRUD (Crear, Leer, Actualizar, Eliminar) de médicos, administradores y jefes de turno.
  + **Módulo de Notificaciones**: Controlador que envía notificaciones a los usuarios (correo o mensajes en la aplicación).
  + **Generador de Reportes**: Módulo que genera informes y métricas sobre las cancelaciones para el dashboard administrativo.

**c. Capa de Persistencia (Base de Datos)**

* **Tecnologías**: MySQL (Amazon RDS).
* **Descripción**: Base de datos relacional donde se almacenará toda la información relevante del sistema.
* **Entidades Principales**:
  + **Usuarios**: Información sobre médicos, administradores y jefes de turno.
  + **Citas**: Detalles de las citas médicas, incluyendo la fecha, hora, médico asignado y estado (cancelada, reprogramada, activa).
  + **Cancelaciones**: Registro histórico de cancelaciones realizadas, incluyendo motivo, fecha y usuario que las realizó.
  + **Notificaciones**: Almacenamiento de notificaciones enviadas y su estado (leídas/no leídas).
  + **Métricas e Informes**: Información generada a partir de las cancelaciones, utilizada para la visualización en el dashboard.

**2. Diagrama de Arquitectura**

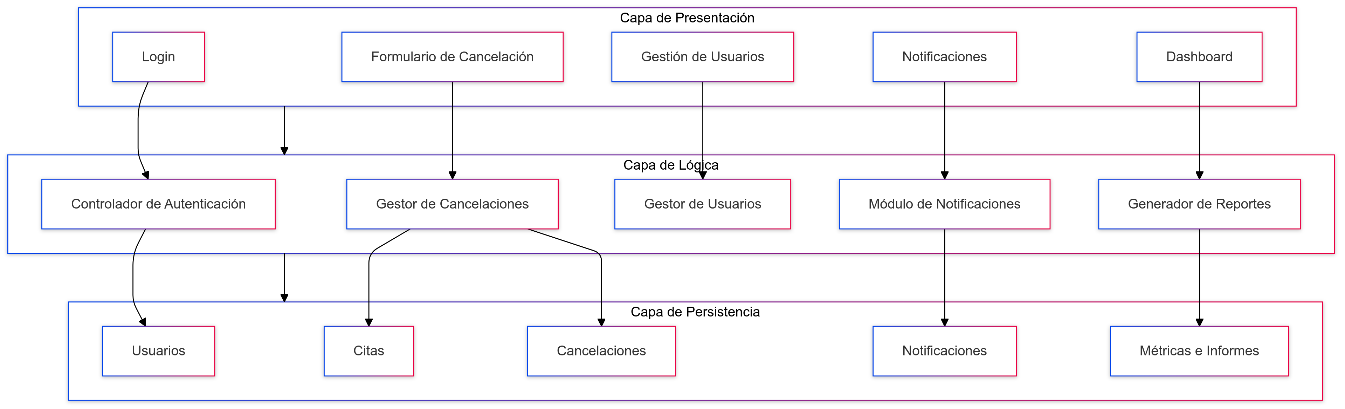
Diagrama

Descripción generada automáticamente

**Flujo de Trabajo del Sistema**

1. **Inicio de Sesión**:
   * El usuario ingresa su RUT y contraseña en la interfaz de React.
   * El Frontend envía una solicitud a la API del Backend para autenticar al usuario.
   * El Backend valida las credenciales y, si son correctas, responde con un token de sesión.
2. **Gestión de Cancelaciones**:
   * El médico accede al formulario de cancelación y envía una solicitud con los datos de la cita que desea cancelar.
   * El Backend valida la solicitud (fecha, motivo) y registra la cancelación en la Base de Datos si es aprobada.
   * Se envían notificaciones automáticas a los usuarios afectados.
3. **Visualización de Métricas**:
   * El administrador accede al dashboard.
   * El Frontend solicita al Backend las métricas y reportes generados sobre las cancelaciones.
   * El Backend consulta la Base de Datos, procesa la información y la devuelve para ser visualizada en el dashboard.

**4. Diagrama de Componentes**



**5. Seguridad**

* **Autenticación y Autorización**: Uso de JWT (JSON Web Tokens) para la autenticación segura de usuarios.
* **Validación de Datos**: Implementación de reglas de negocio en el backend para evitar datos incorrectos.
* **Cifrado de Contraseñas**: Almacenamiento de contraseñas utilizando algoritmos seguros (como bcrypt).

**6. Interfaz de Usuario (UI)**

* **Diseño Responsive**: La interfaz se adaptará a diferentes tamaños de pantalla, optimizando la experiencia para dispositivos móviles y desktop.
* **Estilo**: Uso de los colores y estilos gráficos propios de RedSalud, incluyendo la opción de un **modo oscuro**.

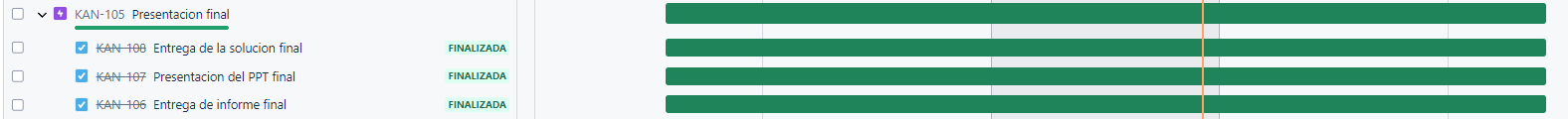
# Metodología

1. **Fases del Proyecto**
2. **Levantamiento de Requerimientos**: Entender necesidades, entrevistas con stakeholders.
3. **Diseño**: Creación de prototipos, diseño de arquitectura y diagramas.
4. **Desarrollo**: Implementación del código en frontend y backend.
5. **Pruebas**: Testeo de la funcionalidad y corrección de errores.
6. **Documentación**: Redacción de manuales, guías y documentación técnica.

**Estructura Gantt**

Imagen que contiene Gráfico de barras

Descripción generada automáticamenteGráfico

Descripción generada automáticamente

**b. Metodología de Gestión del Proyecto: Scrum**

Para la gestión del proyecto, utilizamos la metodología **Scrum**, que nos permitió organizar el trabajo de manera ágil y eficiente. A continuación, se detallan las actividades y roles asignados durante todo el proceso:

**Roles del Equipo Scrum:**

1. **Scrum Master (André Espinoza)**:
   * André lideró las reuniones diarias, facilitó la metodología y se encargó de eliminar cualquier impedimento que pudiera surgir durante los Sprints.
2. **Product Owner**:
   * El Product Owner fue responsable de gestionar el **Product Backlog**, priorizando los requerimientos y asegurándose de que se alinearan con los intereses de los stakeholders.
3. **Equipo de Desarrollo**:
   * El equipo de desarrollo estuvo compuesto por los siguientes miembros:
     + **Javier González**: Encargado de la base de datos y diagramas.
     + **Sebastián Hidalgo**: responsable del desarrollo y las presentaciones.
     + **Matías Rietta**: Desarrollador tanto de frontend como de backend.
     + **Diego Robles**: Encargado de los diagramas y desarrollo de la base de datos.
     + **Matías Sepúlveda**: Desarrollador y encargado de diagramas.

**Fases en Scrum:**

1. **Sprint Planning (Planificación del Sprint)**:
   * En la planificación de cada Sprint, definimos las tareas específicas a realizar, alineando los objetivos con las expectativas de los stakeholders. Cada Sprint tuvo una duración de **2 semanas**.
2. **Daily Scrum (Reunión Diaria)**:
   * Todos los días, el equipo de desarrollo participó en reuniones diarias de **15 minutos** donde se discutieron los avances y se abordaron posibles bloqueos.
3. **Sprint**:
   * Cada Sprint comenzó con la **Planificación** y terminó con la **Revisión** y la **Retrospectiva**. Durante cada Sprint, se completaron las tareas relacionadas con el Levantamiento de Requerimientos, Diseño, Desarrollo, Pruebas y Documentación.
4. **Sprint Review (Revisión del Sprint)**:
   * Al finalizar cada Sprint, se presentó el progreso alcanzado a los stakeholders y se obtuvo retroalimentación para ajustar el curso del proyecto si fuera necesario.
5. **Sprint Retrospective (Retrospectiva del Sprint)**:
   * Después de cada Sprint, el equipo se reunió para reflexionar sobre lo que funcionó bien, lo que se puede mejorar y cómo optimizar la ejecución de los próximos Sprints.

**Product Backlog:**

El **Product Backlog** fue actualizado y priorizado al inicio de cada Sprint. Este contenía todos los elementos necesarios para completar el proyecto, desde la recopilación de requerimientos hasta la entrega final de la documentación. A medida que avanzábamos en los Sprints, el **Product Backlog** fue adaptándose a medida que surgían nuevas necesidades o ajustes a los requisitos iniciales.