

Gestión de proyecto para el desarrollo de sistema de confirmación de cita médica vía mensajería WhatsApp

Alejandro Alberto Reyes Olivares

alejandro.reyeso@usm.cl

Ricardo Cahe
Profesor Guía

Franco Jorquera
Profesor Correferente

Resumen: El presente trabajo aborda la alta tasa de ausentismo de pacientes (No Show) y el consumo intensivo de recursos humanos en la contactabilidad del Hospital Dr. Gustavo Fricke. En la Unidad de Endoscopia, se detectó además una alta cancelación de procedimientos *in situ*, debido a que los pacientes acudían sin la preparación clínica obligatoria.

La solución propuesta fue desarrollar un Producto mínimo viable (MVP) de un Sistema de Confirmación y Envío de Recordatorios. El objetivo fue optimizar la comunicación hospital-paciente, automatizando la entrega de recordatorios de citas y el envío de plantillas con instrucciones de preparación pre-procedimiento mediante la API de WhatsApp.

Para el desarrollo se utilizó Scrum, permitiendo la adaptación continua al flujo de Endoscopia y la refinación de la usabilidad mediante feedback constante del patrocinador, culminando con su evaluación en la Feria de Software.

Los resultados demostraron la capacidad de automatizar el envío masivo de recordatorios, asegurando la entrega de información crítica. El impacto esperado es la reducción significativa del ausentismo y de los procedimientos fallidos, optimizando el uso de recursos y liberando carga administrativa.

Palabras Clave: Metodología Ágil, Recordatorios Automatizados, WhatsApp API, Reducción de ausentismo



CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): ☒ Memoria o trabajo de título ☐ Tesis de Postgrado

Título del trabajo: Gestión de proyecto para el desarrollo de sistema de confirmación de cita médica vía mensajería WhatsApp

Nombre del candidato(a): Alejandro Alberto Reyes Olivares

Carrera / Grado: Ingeniería en informática

Campus: Sede Viña del Mar **Departamento:** Departamento de Electrotecnia e Informática (ELINF)

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Ricardo Cahe, en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución.

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL (marcar una opción)

☒ El trabajo **NO contiene** información que amerite confidencialidad y puede ser publicado de inmediato en repositorio con acceso abierto.

☐ El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (**embargo**) por (**marcar una opción**):

☐ 6 meses ☐ 12 meses ☐ 2 años ☐ 3 años ☐ 5 años ☐ 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 02/07/2026 **Firma:** 

Estudiante o Candidato(a):

Fecha: 02/07/2026 **Firma:** 

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.



Índice

Introducción	3
1.1 Contexto	3
1.2 Planteamiento del problema	3
1.3 Propuesta de solución	4
1.4 Metodologías Ágiles	4
1.5 Objetivos de la tesina	5
1.5.1 Objetivo General	5
1.5.2 Objetivos específicos	5
2 Marco Teórico	6
2.1 Manifiesto Ágil	6
2.2 Scrum	7
2.3 Tecnologías de desarrollo	8
2.4 Herramientas de control de versiones y comunicación	9
2.5 Herramientas de gestión de proyecto	9
3 Metodología y plan de trabajo	9
3.1 Enfoque para la implementación de metodologías ágiles en el proyecto	9
3.2 Herramientas de gestión empleadas	10
3.3 Plan de personal	10
3.4 Planificación temporal del proyecto	11
3.4.1 Primer semestre	11
3.4.2 Segundo semestre	11
4 Planificación de Sprints	12
4.1 Definición de sprints y ciclos de desarrollo	12
4.1.1 Sprint 0	13
4.1.2 Sprint 1	13
4.1.3 Sprint 2	14
4.1.4 Sprint 3	17
4.1.5 FESW	19
4.2 Estimación de tareas y asignación de recursos	21
5 Ejecución de reuniones ágiles	23
5.1 Descripción de reuniones	23
5.2 Colaboración con el patrocinador	23
5.3 Importancia de la comunicación y colaboración en el equipo	23
6 Gestión de cambios y mejoras continuas	24
6.1 Adaptación a cambios en requerimientos	24
6.2 Implementación de mejoras basadas en feedback	24
7 Evaluación del desempeño ágil	25
7.1 Métricas utilizadas para medir la eficiencia y efectividad	25
7.2 Análisis de resultados obtenidos	26
8 Desafíos y soluciones	26



8.1 Identificación de obstáculos y estrategias de mitigación.....	26
9 Estimación de costos	27
9.1 Estimación de costos de recursos humanos.....	27
9.2 Costos operativos.....	28
9.3 Costos de infraestructura y hardware	29
9.4 Costos de licencias de herramientas	29
9.5 Resumen de inversión total	30
9.6 Conclusión económica	30
10 Conclusión.....	31
Referencias	33
Bibliografía.....	33

Lista de Figuras

Figura 1. Cronograma primer y segundo sprint.....	12
Figura 2. Cronograma tercer sprint y FESW.....	13
Figura 3. Incremento Sprint 1 - Vista gestor de citas médicas	15
Figura 5. Incremento Sprint 2 - Vista gestor Usuarios	16
Figura 6. Incremento Sprint 2 - Vista login	17
Figura 7. Incremento Sprint 2 - Mensaje de prueba	17
Figura 8. Incremento Sprint 3 - Gestor de citas médicas.....	19
Figura 9. Incremento Sprint 3 - Gestor de usuarios.....	19
Figura 10. Incremento Sprint 3 - Gestor de médicos	19
Figura 11. Versión final del gestor de citas médicas	21
Figura 12. Versión final del gestor de médicos.....	21
Figura 13. Versión final del gestor de usuarios	22
Figura 14. Versión final de los mensajes	22
Figura 15. Gráfico Velocity	26

Lista de Tablas

Tabla 1. Contexto de requerimientos.....	4
Tabla 2. Roles y responsabilidades.....	11
Tabla 3. Puntuación de Historias de usuario	23
Tabla 4. Rendimiento por ciclo	27
Tabla 5. Costos de recursos humanos	29
Tabla 6. Costos operativos.....	30
Tabla 7. Costos de infraestructura	30
Tabla 8. Costos de herramientas	31
Tabla 9. Valoración de la inversión.....	31

Introducción

1.1 Contexto

A principios del año 2025, en el marco de la asignatura de Proyecto de Software, se plantearon diversos desafíos tecnológicos impulsados por distintas organizaciones. Como equipo, se seleccionó el desafío propuesto por el Hospital Dr. Gustavo Fricke, el cual busca modernizar y optimizar la gestión hospitalaria. En la actualidad, la eficiencia de los centros de salud pública depende en gran medida de la correcta administración de sus agendas médicas. Sin embargo, los métodos tradicionales de contactabilidad han quedado rezagados frente a las nuevas dinámicas de comunicación digital, lo que hace imperativo buscar alternativas tecnológicas que mejoren la interacción entre la institución y sus pacientes para evitar la pérdida de recursos críticos.

1.2 Planteamiento del problema

Actualmente, el Hospital Dr. Gustavo Fricke registra una tasa anual de inasistencias de pacientes (*No-Show*) cercana al 17%. El proceso actual para recordar citas y entregar indicaciones recae en dos métodos ineficientes: la entrega de un comprobante físico (que se extravía con facilidad) y la realización de llamadas telefónicas manuales.

Este último método resulta lento y altamente ineficaz, ya que los pacientes suelen ignorar llamadas de números desconocidos. Esto se debe en parte al aumento sostenido de estafas telefónicas y publicidad no deseada, fenómeno respaldado por estudios recientes que posicionan a Chile como el segundo país de América con mayor cantidad de llamadas *spam* por persona (Hiya, 2024, 2025).

Adicionalmente, en áreas críticas como la Unidad de Endoscopía, esta brecha comunicacional genera un problema doble: los pacientes no solo olvidan su cita, sino que a menudo acuden sin la preparación clínica obligatoria por falta de información oportuna. La suma de estos factores se traduce en procedimientos cancelados in situ, tiempos muertos en la agenda y una grave pérdida de horas médicas e insumos que no pueden ser reasignados.

Requerimiento	Descripción
Gestor local de citas médicas (MVP)	Módulo que permite a los operarios registrar y eliminar los datos de las citas médicas dentro del sistema de notificaciones, gestionar y visualizar estados y ejecutar el envío masivo o individual de mensajes, sin modificar la agenda oficial del hospital.
Mensajería de WhatsApp	Módulo que permite enviar plantillas de mensajes a través de WhatsApp y recibir feedback por webhook.
Gestor de médicos	Módulo que permite a los operarios agregar y eliminar médicos.
Gestor de usuarios y login	Módulo que permite a los operarios iniciar y cerrar sesión, y añadir más usuarios.

Tabla 1. Contexto de requerimientos

1.3 Propuesta de solución

Como respuesta a la problemática expuesta en el punto anterior, este proyecto propone el desarrollo de un Producto Mínimo Viable (MVP) centrado en un Sistema de Confirmación y Envío de Recordatorios. La solución se centra en automatizar la comunicación entre el hospital y el paciente para asegurar no sólo la asistencia, sino la efectividad del procedimiento médico.

El sistema tiene dos propósitos prácticos:

- **Confirmación de asistencia:** Enviar recordatorios automáticos con la fecha y hora de la cita para reducir el ausentismo o No-Show.
- **Envío de instrucciones médicas:** Entregar de forma oportuna las indicaciones de preparación previa (como el ayuno), asegurando que el paciente llegue en las condiciones clínicas adecuadas para que el procedimiento médico no deba ser cancelado en el lugar.

Alcance y límites del proyecto:

- **Independencia del sistema hospitalario:** El proyecto funciona como un sistema aislado para validar la idea. Cuenta con un módulo propio para registrar citas y gestionar los mensajes a nivel local, pero la obtención de datos desde la base de datos oficial del hospital, así como el agendamiento o reagendamiento real de las horas médicas, quedan fuera del alcance.
- **Estado de implementación:** El entregable es un prototipo completamente funcional (MVP) probado en un entorno de desarrollo. El despliegue final y la instalación en los servidores reales del Hospital Dr. Gustavo Fricke no está contemplado en esta etapa.
- **Seguridad y normativas:** Al tratarse de un MVP enfocado en resolver el flujo de comunicación, el endurecimiento avanzado de la ciberseguridad (hardening) y la adaptación estricta a las leyes de protección de datos de pacientes se consideran trabajo futuro. Estos pasos serán esenciales antes de que el hospital decida llevar el sistema a un entorno de producción oficial.

1.4 Metodologías Ágiles

El presente proyecto se desarrolla bajo metodología ágil, guiada por marcos de trabajo como Scrum, que presentan ventajas frente a enfoques tradicionales, tales como:

- **Entrega de valor continua:** Al contar con Sprints de 1 a 4 semanas de duración, se prioriza desarrollar tareas que otorgan valor al usuario y entregar incrementos funcionales.
- **Adaptación al cambio:** A diferencia de los enfoques tradicionales, donde los cambios implican replanificación general, Scrum los incorpora en cada iteración reordenando el Product Backlog y planificando el siguiente Sprint.

- **Menor riesgo y costo del error:** Las revisiones frecuentes e incrementos pequeños permiten detectar errores de manera temprana, haciendo posible su corrección a un menor costo.
- **Entrega de calidad:** Cada incremento se prueba, se revisa y se despliega, lo que reduce fallas y deuda técnica.
- **Mejora continua:** Al final de cada Sprint el equipo realiza la Revisión y la Retrospectiva para evaluar lo entregado y cómo trabajó, fomentando un aprendizaje continuo y mejoras en la forma de trabajar.
- **Colaboración:** Fomenta la comunicación continua entre equipos multidisciplinarios y el cliente; permite ajustar prioridades y alcance con base en evidencia, reduce malentendidos y asegura que el resultado responda a necesidades reales del negocio.
- **Transparencia:** Un tablero de Scrum permite al equipo y a los stakeholders ver el estado de las tareas y el avance real del proyecto.

El uso de metodologías ágiles permitirá al equipo trabajar en iteraciones, favorecer el aprendizaje continuo y el refinamiento constante de la solución, adaptarse a nuevas necesidades y mejorar la calidad del producto.

1.5 Objetivos de la tesina

1.5.1 Objetivo General

Gestionar un proyecto informático utilizando metodologías ágiles para desarrollar de manera eficiente un sistema de recordatorio y confirmación de citas para el Hospital Gustavo Fricke a través de la aplicación de mensajería WhatsApp.

1.5.2 Objetivos específicos

- Gestionar los tiempos de desarrollo para generar incrementos de valor a tiempo en cada iteración.
- Adaptar el marco de trabajo Scrum a un contexto estudiantil para optimizar tiempos para estudio y prácticas profesionales.
- Planificar y asignar tareas considerando las cargas académicas y las prácticas.
- Supervisar el progreso del proyecto para detectar oportunidades de mejora y optimizar el desarrollo.
- Utilizar herramientas de metodologías ágiles para mejorar la comunicación del equipo y facilitar la resolución de problemas.

2 Marco Teórico

En este apartado se describen las herramientas y tecnologías que se utilizarán para la gestión y el desarrollo del proyecto.

2.1 Manifiesto Ágil

El Manifiesto Ágil, creado en el año 2001, establece cuatro valores y doce principios que priorizan la adaptación al cambio, la colaboración y la entrega continua de software funcional.

Valores del Manifiesto Ágil

- **Individuos e interacciones sobre procesos y herramientas:** Las capacidades de comunicación, colaboración y resolución de problemas de los miembros del equipo son esenciales para el éxito del proyecto; los procesos y las herramientas que se utilicen son apoyo.
- **Software funcionando sobre documentación exhaustiva:** Se prioriza entregar software operativo que aporte valor al cliente antes que producir documentación excesiva que potencialmente puede quedar obsoleta a lo largo del desarrollo del proyecto.
- **Colaboración con el cliente sobre negociación contractual:** Se fomenta la participación continua del cliente durante todo el desarrollo para ajustar y mejorar según sus necesidades.
- **Respuesta ante el cambio sobre seguir un plan:** Se valora la capacidad del equipo para aprender y adaptarse a nuevas necesidades en cada iteración.

Principios del Manifiesto Ágil

1. **Satisfacción del cliente:** La mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor real.
2. **Aceptación a los cambios:** Incluso en etapas tardías, se aceptan cambios en los requisitos para proporcionar ventaja competitiva al cliente.
3. **Entregas frecuentes:** Se realizan entregas de productos funcionales en periodos de tiempo lo más cortos posibles, habitualmente en semanas o pocos meses.
4. **Colaboración diaria:** A lo largo de todo el proyecto, tanto miembros del equipo como usuarios colaboran a diario.
5. **Equipos motivados:** Construir proyectos en torno a individuos motivados, brindándoles el entorno, el apoyo, la confianza y la autonomía para decidir cómo llevar a cabo el trabajo.
6. **Comunicación cara a cara:** La forma más eficiente y efectiva de transmitir información hacia y entre miembros del equipo es a través de la conversación cara a cara.
7. **Software funcionando es la medida principal de progreso:** El progreso de un proyecto se mide según las funcionalidades completadas y operativas.
8. **Desarrollo sostenible:** Tanto promotores, miembros del equipo de desarrollo y usuarios deben ser capaces de trabajar a un ritmo constante y evitando el agotamiento.
9. **Atención a la excelencia técnica y buen diseño:** El aprendizaje continuo del equipo permite construir productos de mayor calidad y fáciles de mantener.
10. **Simplicidad:** Maximizar la cantidad de trabajo no realizado a través de construir solo lo necesario.

11. **Equipos autoorganizados:** La colaboración y el conocimiento colectivo favorecen la creación de mejores soluciones.
12. **Reflexión y mejora continua:** De manera regular el equipo realiza reuniones donde reflexionan sobre qué mejorar para ajustar y optimizar su forma de trabajar.

Las metodologías ágiles mejoran la calidad, la satisfacción del cliente y eficiencia al priorizar flexibilidad, colaboración y entrega continua de valor; permiten responder rápido a cambios, adecuarse a las necesidades del cliente y corregir problemas de manera temprana.

2.2 Scrum

Scrum es un marco de trabajo ágil ligero que ayuda a los equipos a generar valor de manera iterativa e incremental; se basa en tres pilares:

- **Transparencia:** Principio fundamental que refiere a la visibilidad de los aspectos del trabajo realizado por el equipo, es decir, toda la información que sea relevante debe ser accesible tanto para los miembros del equipo, como para los stakeholders. Este pilar se centra en la comunicación abierta y sin obstáculos, lo que genera confianza y refuerza la colaboración. Se manifiesta en:
 - **Backlog del Sprint:** Lista dinámica que incluye las tareas que el equipo se compromete a llevar a cabo a lo largo del sprint.
 - **Backlog del producto:** Lista dinámica, ordenada y priorizada que contiene las funciones, requisitos, mejoras y correcciones que aún faltan ser ejecutadas, sirviendo como guía para el equipo.
 - **Revisión del Sprint:** Momento en el que se muestra el trabajo realizado a las partes interesadas, permitiendo su evaluación y generación de feedback.
 - **Definición de hecho (DoD):** Conjunto de criterios claros que definen la finalización de cada tarea.
- **Inspección:** Refiere a la evaluación y revisión regular y frecuente del progreso del proyecto, realizándose a través de reuniones regulares, reuniones diarias y revisiones del sprint; la inspección ayuda al equipo a mantener la trayectoria hacia el éxito del proyecto y permite detectar posibles problemas, desviaciones del plan y tomar medidas correctivas de manera oportuna, lo que fomenta la mejora continua del producto. Se materializa en:
 - **Planificación del Sprint:** Al inicio de cada sprint, el equipo revisa el producto backlog y establece el trabajo que se va a llevar a cabo en el sprint.
 - **Reuniones diarias:** Es un evento con duración de 15 minutos, en el que los miembros del equipo inspeccionan el progreso del sprint, comparten novedades e identifican impedimentos.
 - **Revisión del Sprint:** Al final de cada sprint se realiza una reunión en la que el equipo presenta el trabajo completado a los stakeholders para su revisión, entreguen su opinión y determinar futuras adaptaciones.
 - **Retrospectiva del Sprint:** Luego del término de cada sprint y realizada la revisión, el equipo se reúne para documentar y analizar lo ocurrido en el sprint, identificando lo que hicieron bien, lo que hay que mejorar, y diseñar estrategias de mejora.

- **Adaptación:** Pilar complementario a la inspección. Cuando la inspección revela desviaciones o aprendizajes, el equipo adapta las estrategias y su forma de trabajar con base en los conocimientos adquiridos.
 - **Modificación del backlog de sprint:** En caso de surgir nuevas ideas o que ocurran cambios durante el sprint, el equipo de desarrollo puede modificar el backlog del sprint.
 - **Adaptaciones en las reuniones diarias:** En las reuniones se comunican avances e impedimentos, permitiendo replantear el plan de las próximas 24 horas.
 - **Comentarios durante la revisión del sprint:** El Product Owner realiza adaptaciones en el Product Backlog con base en los comentarios recibidos en la revisión del sprint.

El equipo Scrum está compuesto por:

- **Product Owner:** Se encarga de comunicarse con las partes interesadas, comprende los requisitos y prioriza el Product Backlog para maximizar el valor entregado.
- **Scrum Master:** Es el responsable de que el equipo comprenda y utilice correctamente Scrum; facilita los eventos, ayuda a eliminar impedimentos y protege al equipo de interrupciones externas.
- **Desarrolladores:** Son el equipo encargado de generar un incremento usable en cada sprint; planifican el trabajo en el Sprint Backlog y aseguran calidad según la Definición de hecho (DoD).

2.3 Tecnologías de desarrollo

- **React:** Biblioteca de JavaScript para crear interfaces de usuario basadas en componentes y estado; mejora el renderizado con Virtual DOM y ofrece Hooks para gestionar lógica y efectos.
- **Vite:** Herramienta de desarrollo y compilación front-end con servidor rápido y HMR; genera builds optimizadas para producción.
- **Node.js:** Entorno de ejecución de JavaScript del lado del servidor con modelo asíncrono y orientado a eventos.
- **Express.js:** framework minimalista y flexible para Node.js; proporciona enrutamiento, middleware y utilidades para construir APIs RESTful de forma estructurada.
- **Zod:** Librería de validación de esquemas en tiempo de ejecución para TypeScript/JavaScript; permite definir contratos de datos y garantizar tipos y formatos correctos.
- **PostgreSQL:** Sistema gestor de base de datos relacional (RDBMS) open-source, permite transacciones con atomicidad, consistencia, aislamiento y durabilidad (ACID), con SQL estándar y tipos avanzados como JSONB.
- **Prisma:** ORM para Node.js y TypeScript; incluye Schema, Client y Migrate para definir modelo, realizar consultas type-safe y manejar migraciones de manera controlada.
- **WhatsApp Business (Cloud API):** Plataforma de Meta para enviar y recibir mensajes con plantillas aprobadas, botones e identidad verificada; integra webhooks para estados de entrega.

2.4 Herramientas de control de versiones y comunicación

- Git: Sistema de control de versiones distribuido. Permite trabajar con ramas, historial y etiquetas; realizar revisiones antes del commit; resolver conflictos y revertir cambios con trazabilidad.
- GitHub: Plataforma de alojamiento de repositorios Git con control de versiones, Pull Requests, Issues y Actions para CI/CD; facilita revisión de código y automatización del pipeline.
- Discord: Herramienta de comunicación en tiempo real (texto/voz) con canales, roles y permisos.

2.5 Herramientas de gestión de proyecto

- Jira: Herramienta para seguimiento de trabajo; permite gestionar backlog, planificar sprints, tablero Scrum, calendario, cronograma, creación de reportes y flujos de aprobación.
- Confluence: wiki corporativa para crear y compartir documentación.
- Figma: Plataforma colaborativa de diseño y prototipado; admite componentes y variantes, y permite crear prototipos interactivos.

3 Metodología y plan de trabajo

3.1 Enfoque para la implementación de metodologías ágiles en el proyecto

Para el desarrollo de este proyecto se utilizarán metodologías ágiles, guiadas por el marco de trabajo Scrum. Scrum aportará las herramientas para lograr transparencia, flexibilidad y adaptabilidad. La visualización del avance de cada iteración se realizará mediante el tablero de scrum que proporciona Jira, donde las tareas se representan como tarjetas y se desplazan según su estado, el cual puede ser: "Idea", "To Do", "In Progress", "In Review" y "Done".

El equipo trabajará en sprints de cuatro semanas de duración. Al comienzo de cada iteración se realizará el Sprint Planning, reunión en la que se seleccionarán las tareas a realizar y el objetivo del sprint. Al finalizar, se llevará a cabo la Sprint Review, instancia en la que se evaluará el incremento, se recogerán comentarios de las partes interesadas y se determinarán futuras adaptaciones en caso de ser necesario. Luego de realizada la Sprint Review, se hará una reunión de retrospectiva, donde el equipo analizará el sprint finalizado y acordará un plan de acción para la mejora continua.

Debido a que varios integrantes del equipo realizan su práctica profesional en paralelo con los estudios, no se realizan reuniones diarias. En su lugar, se realizan actualizaciones de manera asíncrona por escrito, de una a dos reuniones presenciales por semana y otras sesiones utilizando herramientas de comunicación en vivo, como Discord.

3.2 Herramientas de gestión empleadas

Para gestionar el proyecto se utilizará Jira. La plantilla Scrum incluye Backlog, Tablero de Scrum, calendario y cronograma. Jira permite planificar y ejecutar Sprints, asignar tareas, priorizar el Product Backlog y generar reportes y métricas.

Para las retrospectivas se utilizará Figma. Utilizando plantillas, notas adhesivas, agrupación por afinidad y votación para priorizar mejoras. Las acciones y la bitácora se documentará en Confluence.

Para el control de versiones y desarrollo de software colaborativo se utilizarán Git y GitHub. Se trabajará con ramas por funcionalidad y Pull Requests para proteger la rama principal.

Para la comunicación síncrona remota, compartir documentos y la realización de algunas reuniones, se utilizará Discord. Se crearán canales de texto específicos por tema para mantener el orden de los tópicos.

3.3 Plan de personal

Esta sección presenta la composición del equipo, sus roles y responsabilidades para la ejecución del proyecto.

Nombre	Rol	Responsabilidades
Martin Chipoco	Product Owner	-Recopilar necesidades del cliente y priorizar el Backlog. -Desarrollo backend. -Integración. -Revisar Pull Requests y realizar los merges.
Alejandro Reyes	Scrum Master	-Facilitar eventos y remover impedimentos. -Desarrollo backend. -Integración.
Vicente Páez	Diseñador UI/UX	-Diseño UI/UX. -Pruebas e investigación de API de WhatsApp.
Ignacio Tenorio	Desarrollador Frontend	-Desarrollo UI React.
Oscar Fernández	Desarrollador Backend	-Desarrollo de API Node/PostgreSQL.
Representante Hospital Gustavo Fricke	Stakeholder	-Participación en Reviews.

Tabla 2. Roles y responsabilidades

3.4 Planificación temporal del proyecto

3.4.1 Primer semestre

Este primer periodo se centra en la planificación inicial del proyecto. Se capturan los requerimientos, se establece la estructura del equipo definiendo tanto roles como responsabilidades, se investigan diversas tecnologías y se selecciona el stack tecnológico a utilizar, y se genera un Product Backlog inicial priorizado.

3.4.2 Segundo semestre

Este segundo periodo corresponde al ciclo de desarrollo, donde en cada iteración los esfuerzos del equipo se concentran en la construcción de un producto mínimo viable (MVP), priorizando la generación de incrementos de valor funcional al término de cada sprint, manteniendo la flexibilidad necesaria para adaptar el alcance del proyecto ante nuevos requerimientos o feedback emergente.

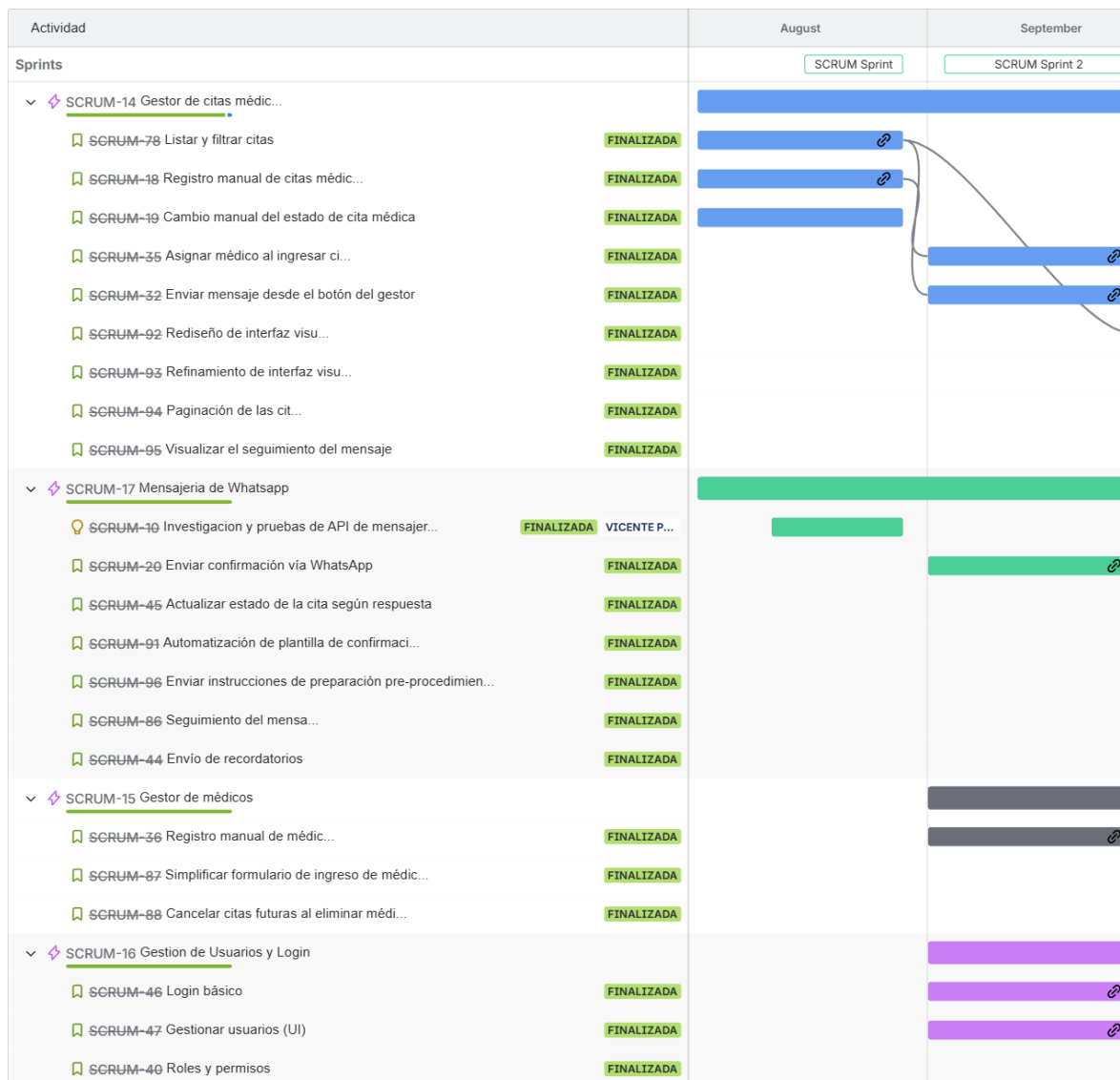


Figura 1. Cronograma primer y segundo sprint

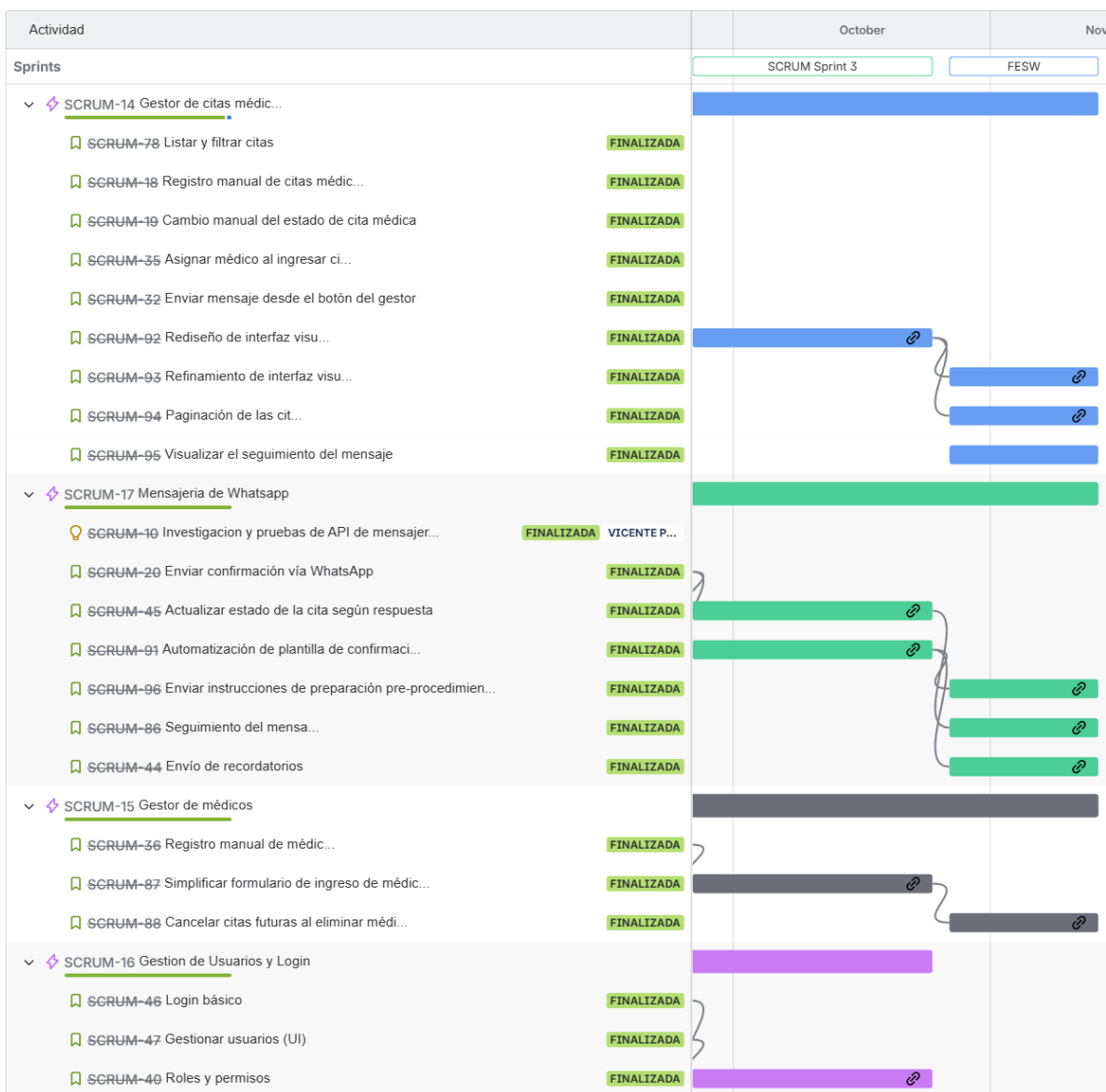


Figura 2. Cronograma tercer sprint y FESW

4 Planificación de Sprints

4.1 Definición de sprints y ciclos de desarrollo

Un sprint es un período de tiempo, típicamente de dos a cuatro semanas en el que el equipo Scrum crea un incremento usable. Para el desarrollo de este proyecto se cuenta con sprints de cuatro semanas de duración, en cada ciclo se eligen y priorizan las tareas que se desarrollarán, así como también se define el objetivo del sprint.

Aunque el marco de trabajo Scrum tradicional no asigna fechas de inicio y término rígidas a las tareas individuales dentro del ciclo de desarrollo, el equipo optó por incluir

estimaciones temporales a modo de guía procedimental y visual. Esta decisión respondió a la necesidad de gestionar la disponibilidad asimétrica de los integrantes. En la práctica, estas fechas no se utilizaron como plazos prescriptivos o restrictivos, sino como una herramienta diagnóstica y un mecanismo de alerta temprana. Si la representación visual indicaba que una tarea excedía su ventana de tiempo estimada, esto operaba como el indicador necesario para activar la redistribución dinámica de la carga de trabajo hacia otro integrante con mayor disponibilidad, garantizando así la fluidez del desarrollo sin romper la autoorganización del equipo.

4.1.1 Sprint 0

Al final del primer semestre se realiza una iteración de inicio llamada "Sprint 0". Aunque no forme parte oficial de Scrum y, que generalmente no genere ningún incremento, permite al equipo en un periodo de dos semanas prepararse para dar comienzo al desarrollo del proyecto, definiendo diferentes puntos tales como:

- Definir la visión del producto, identificar los stakeholders y establecer las métricas de éxito.
- Construir el Product Backlog inicial, priorizado y con criterios de aceptación.
- Evaluar y seleccionar las herramientas y tecnologías a utilizar.
- Formalizar la estructura del equipo, asignando roles y responsabilidades.

4.1.2 Sprint 1

Al inicio del segundo semestre comienza el Sprint 1, el cual tiene como objetivo crear el apartado visual del gestor de citas médicas. Debido a ser una etapa temprana del proyecto, la mayoría de los esfuerzos están destinados al frontend, diseñando una interfaz fácil de usar y desarrollando solo funcionalidades que permitan visualizar y modificar datos cargados en memoria caché. En paralelo, se investigan alternativas para enviar mensajes a través de WhatsApp, probando Ultramsg y WhatsApp Cloud API.

Historias de usuarios:

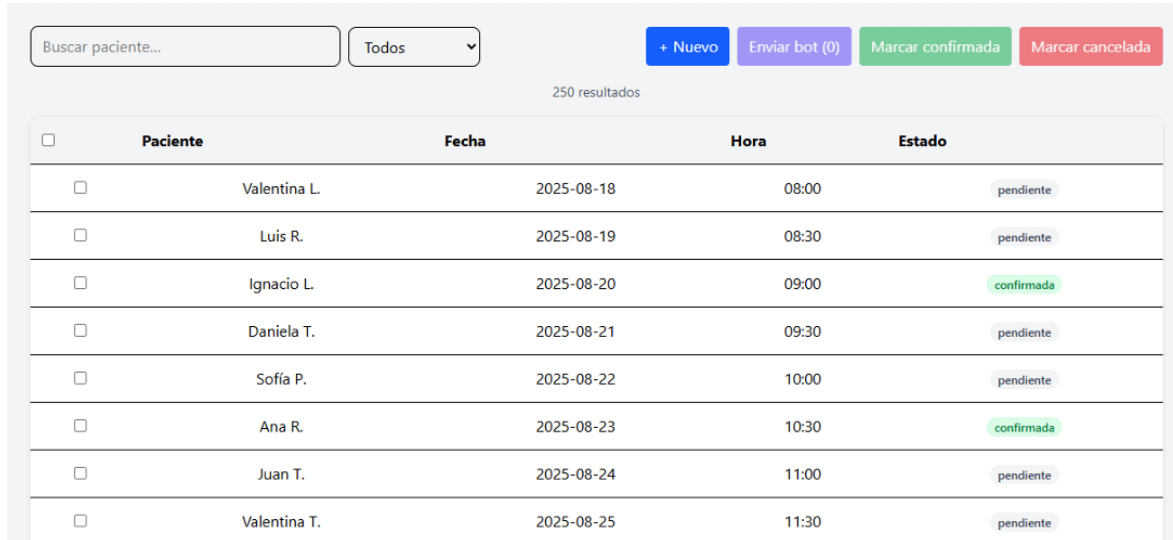
- Listar y filtrar citas: Como operario quiero visualizar y filtrar las citas médicas, para encontrar rápidamente la información de un paciente específico o ver las citas de un estado específico.
- Registro manual de citas médicas: Como operario quiero registrar manualmente una nueva cita, para ingresar al sistema los pacientes a los que debo contactar.
- Cambio manual del estado de cita médica: Como operario quiero cambiar el estado de una cita de manera manual, para modificarla en caso de surgir un imprevisto.

Spike:

- Investigación y pruebas de API de mensajería: Se investigaron y probaron diversas APIs que permiten automatizar el envío de mensajes a través de WhatsApp, resultando en la elección de la API oficial de Meta.

Tareas:

- Creación de la vista del gestor de citas (04/08 - 14/08).
- Llenado y filtrado de tabla de citas (15/08 - 19/08).
- Registro de cita (19/08 - 22/08).
- Cambio manual del estado(25/08 - 27/08).



☐	Paciente	Fecha	Hora	Estado
☐	Valentina L.	2025-08-18	08:00	pendiente
☐	Luis R.	2025-08-19	08:30	pendiente
☐	Ignacio L.	2025-08-20	09:00	confirmada
☐	Daniela T.	2025-08-21	09:30	pendiente
☐	Sofía P.	2025-08-22	10:00	pendiente
☐	Ana R.	2025-08-23	10:30	confirmada
☐	Juan T.	2025-08-24	11:00	pendiente
☐	Valentina T.	2025-08-25	11:30	pendiente

Figura 3. Incremento Sprint 1 - Vista gestor de citas médicas

4.1.3 Sprint 2

El segundo sprint tiene como objetivo implementar la base de datos y desarrollar las funcionalidades que permitan guardar, leer, eliminar, procesar y visualizar los datos de los diversos gestores, y habilitar el envío de un mensaje de prueba por WhatsApp. En paralelo, se desarrollan las interfaces visuales de los gestores de médicos y de usuarios, así como el inicio de sesión.

Historias de usuarios:

- Asignar médico al ingresar la cita: Como operario quiero que los datos del médico se rellenen de manera sencilla a la cita, para disminuir el tiempo que me demoro en ingresar una nueva cita.
- Enviar mensaje desde el botón del gestor: Como operario quiero que el sistema me permita contactar al paciente de manera sencilla desde el gestor de citas, para facilitar el flujo de trabajo.
- Enviar confirmación vía WhatsApp: Como operario quiero que el sistema se contacte con los pacientes vía WhatsApp, para confirmar su asistencia a la cita médica agendada y ahorrar tiempo en llamadas telefónicas.
- Registro manual de médicos: Como operario quiero ingresar manualmente los médicos en el sistema, para que queden disponibles para ser asignados a las citas médicas.

- Login básico: Como usuario del sistema quiero iniciar sesión ingresando mis credenciales, para proteger la confidencialidad de la información de los pacientes.
- Gestionar usuarios: Como administrador quiero gestionar a los usuarios habilitados desde un panel de administración, para mantener el control sobre qué personas del equipo tienen acceso al sistema.

Tareas:

- Creación del modelo de cita médica y agregado a la base de datos (01/09).
- Creación del modelo de médico y agregado a la base de datos (01/09).
- Desarrollo del controlador del gestor de citas (03/09 - 16/09).
- Creación de los endpoints de citas (04/09 - 09/09).
- Creación de los validadores de solicitudes de citas (10/09 - 15/09).
- Construcción de contratos de datos de citas (15/09 - 17/09).
- Autorrellenado de datos del médico en creación de cita (18/09 - 19/09).
- Creación de plantilla de mensaje de confirmación (01/09).
- Creación de conexión con API de WhatsApp (02/09 - 12/09).
- Envío de plantilla de confirmación (15/09 - 16/09).
- Desarrollo de la vista del gestor de médicos (02/09 - 08/09).
- Desarrollo del controlador del gestor de médicos (03/09 - 08/09).
- Creación de endpoints del gestor de médicos (05/09 - 08/09).
- Construcción de contratos de datos de médicos (09/09 - 15/09).
- Creación de los validadores de solicitudes de médicos (15/09 - 17/09).
- Desarrollo de la vista del gestor de usuarios (10/09 - 15/09).
- Desarrollo de la vista del login (16/09 - 17/09).

Panel del Sistema Administrador (admin) [Cerrar sesión](#)

Citas Usuarios **Médicos**

Médicos

Nombre

Especialidad

Email

Teléfono

Añadir

Nombre	Especialidad	Email	Teléfono	Estado	Acciones
Alejandro Reyes	Medico General			activo	Eliminar
Ignacio Tenorio	Oftalmologo			activo	Eliminar

Figura 4. Incremento Sprint 2 - Vista gestor médicos

Panel del Sistema Administrador (admin) [Cerrar sesión](#)

Citas Usuarios **Médicos**

Usuarios operarios

+ Nuevo

ID	Usuario	Nombre	Rol	Activo	Creado	Acciones
1	operario1	Operario Uno	Operario	Activo	04-09-2025, 2:57:56 p. m.	Editar Eliminar
2	operario2	Operario Dos	Operario	Activo	04-09-2025, 2:57:56 p. m.	Editar Eliminar

Figura 5. Incremento Sprint 2 - Vista gestor Usuarios



Iniciar sesión

Usuario

admin

Contraseña

admin123

Ingresar

Figura 6. Incremento Sprint 2 - Vista login

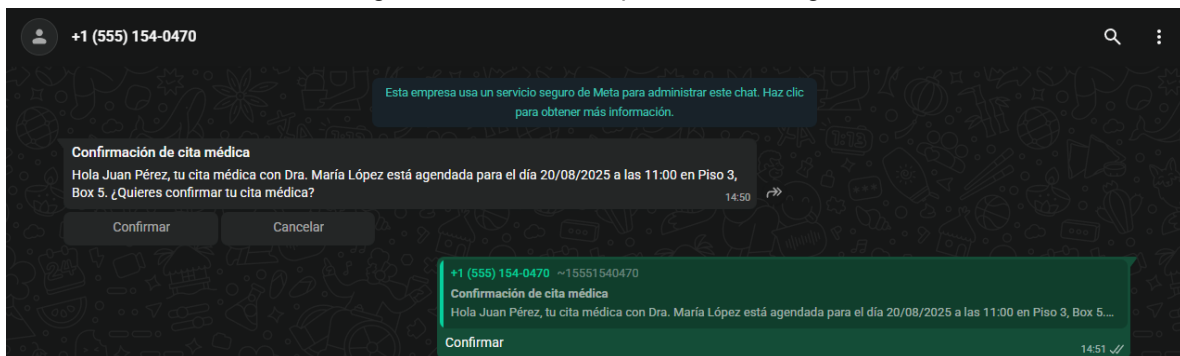


Figura 7. Incremento Sprint 2 - Mensaje de prueba

4.1.4 Sprint 3

Debido a cambios en los requerimientos y comentarios recibidos en la revisión del segundo sprint, el objetivo de la tercera iteración fue rediseñar la interfaz visual del sistema y direccionar el funcionamiento del sistema al área de Endoscopia del hospital. Se desarrolló un webhook y un túnel para recibir las respuestas provenientes de Cloud API; se crearon funciones que procesan lo que recibe el webhook, enlazando las respuestas a la cita médica y cambiando su estado visual de manera automática según lo respondido por el paciente (pudiendo ser "confirmada" o "cancelada", "pendiente" en caso de no recibir respuesta aún).

Historias de usuarios:

- Rediseño de interfaz visual: Como operador del sistema quiero una interfaz visual rediseñada con mejor distribución, colores y más intuitiva, para mejorar la apariencia y usabilidad del sistema.
- Actualizar estado de la cita según respuesta: Como operario quiero que el sistema actualice automáticamente el estado de la cita al recibir la respuesta del paciente, para no tener que cambiarlo manualmente y mantener la agenda actualizada en tiempo real.
- Automatización de plantilla de confirmación: Como operario quiero que la plantilla de confirmación se autocomplete con los datos de la cita, para agilizar el envío de mensajes.
- Simplificar formulario de ingreso de médicos: Como administrador quiero un formulario de registro de médicos simplificado que solicite solo los datos esenciales, para disminuir el tiempo de ingreso de nuevos médicos al sistema.
- Roles y permisos: Como administrador quiero asignar roles y permisos a cada usuario, para asegurar que cada integrante del equipo acceda únicamente a las funciones correspondientes a su cargo.

Tareas:

- Crear nuevo prototipo del apartado visual (29/09 - 03/10).
- Implementación del apartado visual (06/10 - 14/10).
- Cambio de estado según respuesta (16/10 - 20/10).
- Creación de tabla "mensajes" en la base de datos (29/09).
- Creación del Webhook para la API de WhatsApp (29/09 - 10/10).
- Volver dinámica la plantilla de confirmación (14/10 - 15/10).
- Autollenado de la plantilla de confirmación (16/10 - 17/10).
- Creación de plantillas para endoscopia (21/10).
- Eliminación de campos innecesarios del gestor de médicos (14/10 - 15/10).
- Eliminación de atributos innecesarios de los validadores y controladores (15/10 - 17/10).
- Creación del funcionamiento básico del gestor de usuarios (29/09 - 06/10).



Panel del Sistema

Administrador (admin) [Cerrar sesión](#)

Citas

Usuarios

Médicos

Buscar por paciente, RUT o teléfono...

Todos los estados

Todas las especialidades

☐ Mostrar RUT

+ Nuevo

Enviar bot (0)

8 resultados

☐	Paciente	Fecha	Médico	Especialidad	Teléfono	Estado
☐	Juan Carlos Bodoque	2025-09-22T12:00:00.000Z	Ignacio Tenorio	Oftalmologo	955333737	cancelada
☐	juan	2025-10-08T23:30:00.000Z	Ignacio Tenorio	Oftalmologo	955333737	confirmada
☐	Martin Chipoco	2025-10-24T14:00:00.000Z	Alejandro reyes	Endoscopia	979685391	cancelada
☐	Alejandro Reyes	2025-10-24T15:00:00.000Z	Alejandro reyes	Endoscopia	955333737	confirmada
☐	Oscar Fernández	2025-10-24T16:00:00.000Z	Alejandro reyes	Endoscopia	973882698	confirmada
☐	juanito	2025-10-24T18:15:00.000Z	JUAN	Cardiologia	973882698	pendiente
☐	Vicente paez	2025-10-25T12:00:00.000Z	Alejandro reyes	Endoscopia	966484260	cancelada
☐	juan	2025-11-17T02:16:00.000Z	Alejandro reyes	Endoscopia	985776097	pendiente

Confirmar

Cancelar

Eliminar seleccionadas

Figura 8. Incremento Sprint 3 - Gestor de citas médicas

Panel del Sistema

Administrador (admin) [Cerrar sesión](#)

Citas

Usuarios

Médicos

Usuarios operarios

Buscar por nombre, usua

+ Nuevo

ID	Usuario	Nombre	Rol	Activo	Creado	Acciones
1	operario1	Operario Uno	Operario	Activo	04-09-2025, 2:57:56 p. m.	Editar Eliminar
2	operario2	Operario Dos	Operario	Activo	04-09-2025, 2:57:56 p. m.	Editar Eliminar

Figura 9. Incremento Sprint 3 - Gestor de usuarios

Panel del Sistema

Administrador (admin) [Cerrar sesión](#)

Citas

Usuarios

Médicos

Médicos

Buscar por nombre, especialidad, email...

+ Agregar

Nombre	Especialidad	Estado	Acciones
Alejandro reyes	Endoscopia	activo	Eliminar
Alejandro Reyes	Medico General	inactivo	Eliminar
asdasdd	saddas	activo	Eliminar
carlos	endoscopia	activo	Eliminar
dsa	dasd	inactivo	Eliminar
Ignacio Tenorio	Oftalmologo	inactivo	Eliminar
JUAN	Cardiologia	activo	Eliminar
juanito	cardiologia	inactivo	Eliminar
juanito	cardiologia	inactivo	Eliminar

Figura 10. Incremento Sprint 3 - Gestor de médicos

El objetivo de este último ciclo es refinar aspectos visuales, añadir paginación a las citas médicas, integrar nuevas plantillas de mensajes y recordatorios orientados al área de endoscopia, y brindar más retroalimentación al funcionario sobre el estado del mensaje, siendo añadidos más estados ("Enviado", "Recibido", "Leído" y "Recordado")

Historias de usuarios:

- Refinamiento de interfaz visual: Como operario quiero un refinamiento en la distribución visual de los elementos, para operar el sistema de manera más fluida.
- Paginación y orden de las citas: Como operario quiero visualizar el listado de citas dividido y ordenado en páginas numeradas, para navegar de manera fluida y organizada cuando existan grandes volúmenes de citas.
- Visualizar el seguimiento del mensaje: Como operario quiero visualizar el estado de seguimiento del mensaje de manera directa en el gestor de citas, para conocer en tiempo real si el paciente recibió la información.
- Enviar instrucciones de preparación pre-procedimiento: Como operario quiero que el sistema envíe automáticamente las indicaciones de preparación, para asegurar que el paciente llegue correctamente preparado para el examen y evitar cancelaciones de último momento.
- Seguimiento del mensaje: Como operario quiero obtener trazabilidad del mensaje, para confirmar que el paciente efectivamente leyó la información y tomar acciones alternativas en caso de no recibir respuesta.
- Envío de recordatorios: Como operario quiero que el sistema envíe recordatorios de manera automática antes de la cita, para reducir la tasa de ausentismo.
- Cancelar citas futuras al eliminar médico: Como administrador quiero que el sistema cancele automáticamente las citas futuras de un médico al eliminarlo, para mantener consistencia en el gestor de citas.

Tareas:

- Paginación de citas (Backend) (27/10 - 30/10).
- Paginación de citas (Frontend) (30/10 - 03/11).
- Ordenado de citas (27/10 - 31/10).
- Refinar colores de la interfaz visual (30/10 - 31/10).
- Refinar base de datos (03/11 - 04/11).
- Recargado automático de la tabla de citas (04/11 - 05/11).
- Agregar los nuevos estados a los validadores (06/11).
- Corrección de bug en la hora de las citas (07/11 - 10/11).
- Corrección de bug en la selección de citas al enviar mensaje (07/11 - 10/11).
- Diseñar píldoras para los nuevos estados (09/11).
- Integrar nuevas píldoras al gestor de citas (10/11).
- Actualizar la plantilla de confirmación de cita (28/10).
- Envío de plantilla de cancelación de cita (28/10 - 29/10).
- Envío de plantilla de preparativos pre-procedimiento (29/10).
- Modularizar el envío de plantillas (30/10 - 03/11).
- Obtención de los estados (03/11 - 05/11).
- Bloqueo del estado una vez respondido (03/11 - 06/11).
- Envío de recordatorios automatizado (06/11 - 11/11).
- Cambio de estado automático de citas al eliminar médico (03/11 - 11/11).

 Citas Médicos Usuarios Administrador · admin Salir

Buscar por paciente o teléfono... Todos los estados Todas las especialidades + Nuevo Enviar bot (0)

Eliminadas 1 cita(s)

☐	Paciente	Fecha	Médico	Especialidad	Teléfono	Estado
☐	OSCAR FERNÁNDEZ	05/12/2025 11:54	Juan Carlos Bodoque	Endoscopia	973882698	Cancelada
☐	ALEJANDRO REYES	05/12/2025 13:35	Juan Carlos Bodoque	Endoscopia	955333737	Pendiente
☐	Alejandro Reyes	09/11/2025 00:00	Juan Carlos Bodoque	Endoscopia	955333737	Confirmada
☐	Juanito	10/11/2025 23:00	Juan Carlos Bodoque	Endoscopia	955333737	Confirmada
☐	ALEJANDRO	18/11/2025 02:00	Juan Carlos Bodoque	Endoscopia	955333737	Confirmada
☐	Vicente Paez	18/11/2025 15:50	Juan Carlos Bodoque	Endoscopia	966484260	Confirmada
☐	Test 11	19/11/2025 00:29	Juan Carlos Bodoque	Endoscopia	955333737	Recordado
☐	MARTIN CHIPOCO	19/11/2025 19:43	Juan Carlos Bodoque	Endoscopia	979685391	Recibido
☐	VICENTE PAEZ	19/11/2025 20:44	Juan Carlos Bodoque	Endoscopia	966484260	Recordado
☐	Ignacio Tenorio	21/11/2025 13:49	Juan Carlos Bodoque	Endoscopia	985776097	Leído

Mostrando 1-10 de 13 « Anterior Página 1 de 2 Siguiente » 10 / pág

Confirmar Cancelar Eliminar seleccionadas

Figura 11. Versión final del gestor de citas médicas

 Citas Médicos Usuarios Administrador · admin Salir

Médicos

Buscar por nombre, especialidad, email... ☐ Ver inactivos + Agregar

Nombre	Especialidad	Estado	Acciones
Juan Carlos Bodoque	Endoscopia	activo	Habilitar Deshabilitar

Figura 12. Versión final del gestor de médicos

 Citas Médicos Usuarios Administrador · admin Salir

Usuarios operarios

Buscar por nombre, usu: + Nuevo

ID	Usuario	Nombre	Rol	Activo	Creado	Acciones
1	operario1	Operario Uno	Operario	Activo	09-11-2025, 11:53:05 p. m.	Editar Eliminar
2	operario2	Operario Dos	Operario	Activo	09-11-2025, 11:53:05 p. m.	Editar Eliminar

Figura 13. Versión final del gestor de usuarios

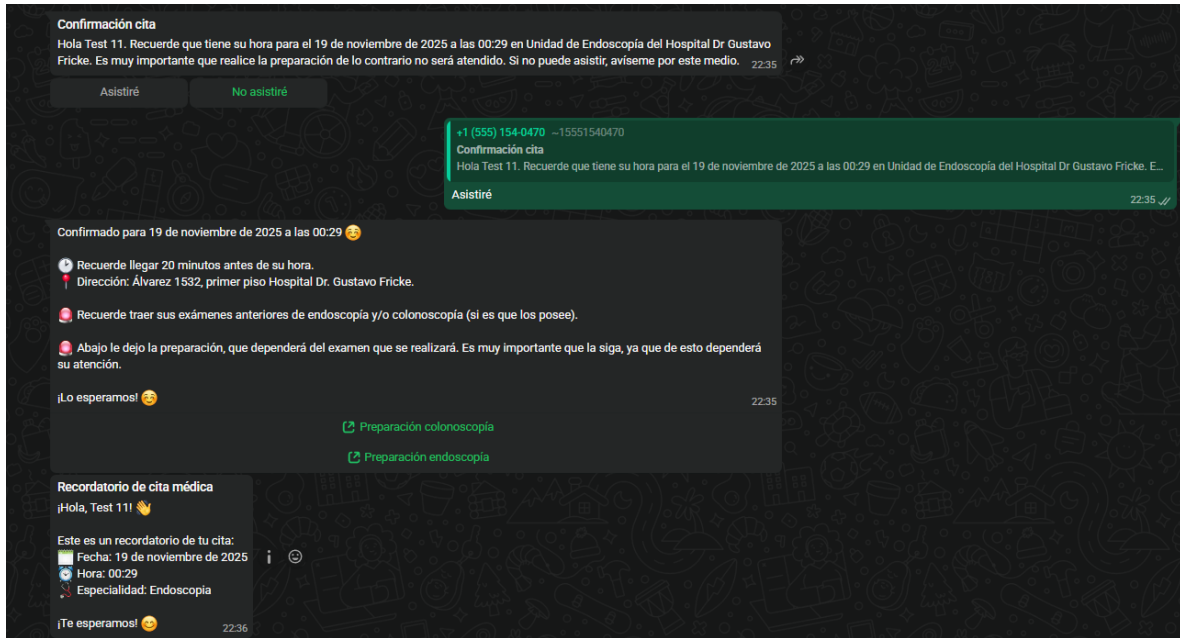


Figura 14. Versión final de los mensajes

4.2 Estimación de tareas y asignación de recursos

Para la estimación de la complejidad de los requerimientos, se hará uso de puntos de historia, método que evalúa el esfuerzo considerando factores como la complejidad técnica, la incertidumbre y el volumen de trabajo. La asignación de los puntos sigue la sucesión de Fibonacci (1,2,3,5,8,13), donde una mayor puntuación corresponde a una mayor complejidad.

Para mitigar cuellos de botella en el desarrollo del proyecto y asegurar el cumplimiento del objetivo del sprint, se aplicará una estrategia de redistribución de tareas basada en la disponibilidad del equipo.

Epica	Historias de usuario	Puntos asignados
Gestor de citas médicas	Listar y filtrar citas	13
	Registro manual de citas médicas	8
	Cambio manual del estado de cita	5
	Enviar mensaje desde el botón	2

	Rediseño de interfaz visual	8
	Refinamiento de la interfaz visual	5
	Paginación de las citas	5
	Visualizar el seguimiento del mensaje	3
Mensajería de WhatsApp	Investigación y pruebas de API de mensajería	8
	Enviar confirmación vía WhatsApp	3
	Actualizar estado de la cita según respuesta	8
	Automatización de plantilla de confirmación	3
	Enviar instrucciones de preparación	5
	Seguimiento del mensaje	3
	Envío de recordatorios	5
Gestor de médicos	Registro manual de médicos	8
	Simplificar formulario de ingreso	2
	Cancelar citas futuras al eliminar médico	5
Gestión de usuarios y login	Login básico	3
	Gestionar usuarios	5
	Roles y permisos	3
	Puntos totales	110

Tabla 3. Puntuación de Historias de usuario

5 Ejecución de reuniones ágiles

5.1 Descripción de reuniones

Debido a la naturaleza académica y a que algunos integrantes se encuentran realizando su práctica profesional a la par, se opta por adaptar algunas de las ceremonias de Scrum, realizándose de la siguiente manera:

- **Planificación del Sprint:** Este evento marca el inicio de cada ciclo de trabajo. Se lleva a cabo el primer día hábil desde el comienzo del sprint. En esta sesión, el equipo define el objetivo del sprint y selecciona los ítems del Backlog a desarrollar.
- **Reuniones diarias adaptadas:** Para garantizar la comunicación efectiva entre los miembros del equipo, monitorear el progreso e identificar bloqueos, estas reuniones se realizan de diversos modos:
 - Presencial: Se realizan de una a dos reuniones físicas por semana.
 - Virtual Síncrono: Se realizan de una a dos sesiones semanales utilizando la plataforma Discord.
 - Virtual Asíncrono: Se mantienen actualizaciones constantes mediante la utilización de canales de texto tanto de Discord como de WhatsApp.
- **Reuniones de resolución de dudas:** De manera complementaria a las reuniones agendadas, se realizan reuniones espontáneas para resolver dudas o bloqueos. En este tipo de reuniones no es necesario que participe todo el equipo, sino que participan únicamente los involucrados en la problemática.
- **Revisión del Sprint:** Esta ceremonia se efectúa acorde al calendario académico. Su objetivo es inspeccionar el incremento generado en el Sprint, presentando el avance tanto a profesores, pares académicos y patrocinadores para obtener feedback.
- **Retrospectiva:** Esta reunión se efectúa entre uno a dos días luego de realizada la revisión del sprint. Utilizando Discord y Figma, el equipo analiza su forma de trabajar para identificar qué funcionó bien y qué debe mejorar en las siguientes iteraciones.

5.2 Colaboración con el patrocinador

La interacción con el patrocinador del proyecto es constante y estructurada. Además de las revisiones formales acordes al calendario académico, se agendan reuniones extraoficiales con el fin de resolver dudas puntuales y obtener información adicional acerca del hospital.

5.3 Importancia de la comunicación y colaboración en el equipo

La comunicación y colaboración son un pilar fundamental para el éxito del proyecto, manteniendo una interacción constante y fluida se logra mantener al equipo alineado con la visión del proyecto, garantizando el cumplimiento de los objetivos de los sprints. Adicionalmente, permite la detección temprana de obstáculos y la reducción del

retrabajo al minimizar malentendidos sobre los requerimientos. Gracias a una colaboración continua se logra una mayor eficiencia en el desarrollo y facilita la adaptabilidad del producto a las necesidades reales del negocio.

6 Gestión de cambios y mejoras continuas

La capacidad de adaptación a los cambios es un factor fundamental para lograr el éxito del proyecto, especialmente en entornos donde los requerimientos y el conocimiento evolucionan constantemente. Este enfoque permite al equipo integrar aprendizajes continuamente e implementar ajustes en los requerimientos de manera eficiente, asegurando que el producto se alinee a las necesidades reales del negocio.

6.1 Adaptación a cambios en requerimientos

La mejora continua es un pilar clave de la filosofía ágil, sustentándose en la recopilación de retroalimentación tanto externa como interna. Este mecanismo permite perfeccionar de manera incremental tanto el producto como la forma de trabajar del equipo.

Las principales fuentes de retroalimentación y aprendizaje son:

- **Validación con el patrocinador:** Las revisiones periódicas actúan como un filtro de calidad y las observaciones recibidas en estas instancias se convierten en nuevos ítems del backlog para llevarse a cabo en el siguiente sprint, asegurando que el producto no se desvíe de las necesidades reales del hospital.
- **Retrospectivas:** La realización de esta ceremonia al final de cada sprint contribuye a la mejora continua del equipo. En ella, no solo se analizan los entregables, sino que también la forma de trabajar del equipo. Gracias a estas instancias, se logran identificar oportunidades de mejora y se definen acciones para integrarlas en el ciclo siguiente.

6.2 Implementación de mejoras basadas en feedback

Durante el desarrollo del proyecto surgieron diversos cambios en los requerimientos que llevaron a redefinir aspectos técnicos y funcionales tales como:

- **Cambio de Arquitectura de datos (Sprint 1):** Luego de obtener información más detallada sobre la infraestructura del hospital, se determinó la inviabilidad de la solución NoSQL propuesta, llevando al cambio hacia una base de datos relacional.
- **Refinamiento de la interfaz de usuario (Sprint 3):** En la revisión del sprint 2, se solicitó rediseñar el apartado visual del sistema, para lograr una mejor usabilidad y estética.
- **Redirección del alcance hacia Endoscopia (Sprint 3 y FESW):** Al inicio del sprint 3 se recibe la solicitud de enfocar el funcionamiento del sistema hacia la unidad de Endoscopia. Esta modificación implicó una simplificación de la arquitectura del sistema, ajuste en las plantillas de mensajes y la modificación del backend para acomodarse al funcionamiento del nuevo flujo de negocio.

- **Incorporación de trazabilidad por descubrimiento técnico (FESW):** Durante el desarrollo del apartado de mensajería, se detectó la oportunidad de obtener mayor trazabilidad de los mensajes. Se integró la funcionalidad de obtención de estados de entrega (Enviado, Recibido, Leído), logrando entregar más información sobre el mensaje al operario del sistema.

7 Evaluación del desempeño ágil

Evaluar el desempeño del equipo es clave para entender cómo funcionó el equipo durante el desarrollo. Para lograrlo, se analiza la cantidad de trabajo realizado frente a lo planificado en cada ciclo, permitiendo visualizar la evolución de la productividad.

7.1 Métricas utilizadas para medir la eficiencia y efectividad

Para lograr una visión objetiva sobre el rendimiento, se utilizaron dos indicadores principales:

- **Velocidad del equipo (Velocity):** Mide la cantidad de trabajo que el equipo logró terminar en cada sprint a través de los puntos de historia. Su objetivo es ayudar a entender el ritmo de trabajo para mejorar la planificación de sprints futuros, prediciendo con mayor exactitud la cantidad de trabajo que puede llevar a cabo el equipo.

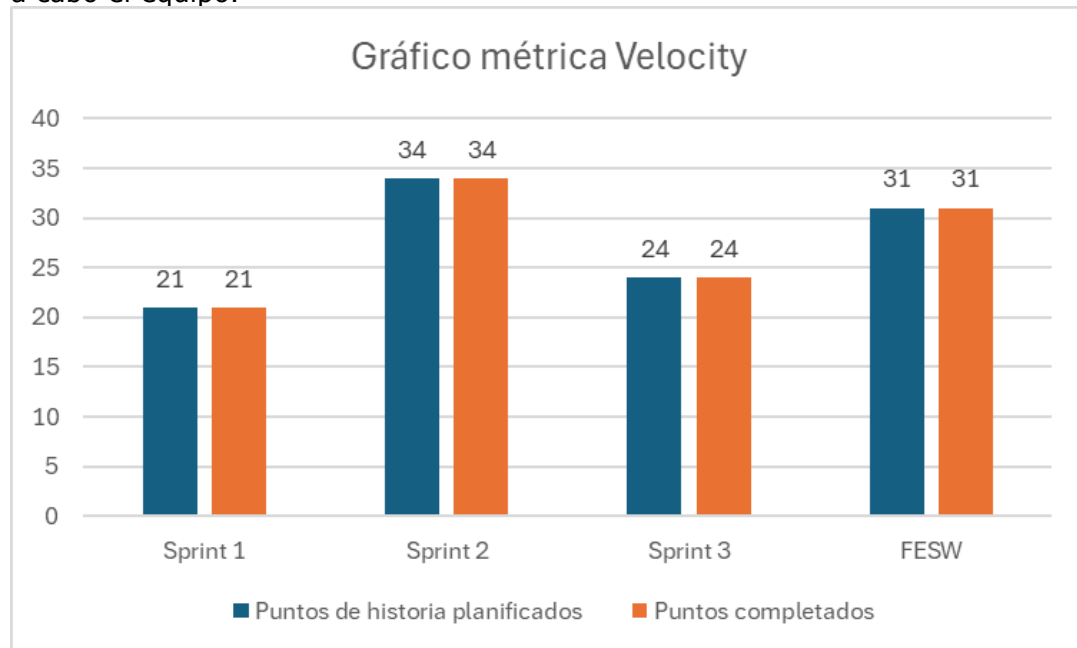


Figura 15. Gráfico Velocity

- **Porcentaje de cumplimiento:** Mide porcentualmente la variación entre los puntos que el equipo se comprometió a desarrollar y los que finalmente se

completaron. Este indicador evalúa la fiabilidad de la planificación, indicando qué tan precisas fueron las estimaciones.

Ciclo	Puntos planificados	Puntos completados	Porcentaje de cumplimiento
Sprint 1	21	21	100%
Sprint 2	34	34	100%
Sprint 3	24	24	100%
FESW	31	31	100%

Tabla 4. Rendimiento por ciclo

7.2 Análisis de resultados obtenidos

El equipo demostró una alta predictibilidad, logrando un 100% de cumplimiento en el desarrollo de los puntos de historia planificados en todos los ciclos, validando así, la efectividad del proceso de estimación y de las estrategias internas para redistribuir las tareas. Esta capacidad de ajuste fue vital para absorber la carga externa altamente variable de los integrantes, asegurando que el desarrollo del MVP mantuviera su ritmo constante sin comprometer los objetivos de cada ciclo.

El proyecto logró finalizar cumpliendo la totalidad de las funcionalidades principales requeridas para esta etapa. La única tarea pendiente se identifica como una labor de deuda técnica: la limpieza de campos y tablas que quedaron en desuso en la base de datos durante el proceso de desarrollo iterativo. Es importante señalar que este punto es meramente estructural y no impacta en la operatividad del sistema ni resta valor directo al patrocinador, quedando como una tarea de mantenimiento para futuras versiones del software.

8 Desafíos y soluciones

Durante el desarrollo del proyecto, el equipo se enfrentó a diversos obstáculos relacionados con el tiempo, las tecnologías y los cambios solicitados por el patrocinador. Para superarlos, no se buscó eliminar los problemas, sino adaptarse rápidamente para continuar con el trabajo.

8.1 Identificación de obstáculos y estrategias de mitigación

- **Desafíos con el tiempo y la comunicación:** Debido a que varios integrantes del equipo realizaban su práctica profesional al mismo tiempo, existió una disparidad en los tiempos disponibles. Además, al inicio se detectaron problemas con la comunicación, lo que a veces llevó a que surgieran malinterpretaciones sobre lo que se debía desarrollar.
- **Estrategias de mitigación:**

- **Redistribución de tareas:** Cuando un integrante se vio superado por la falta de tiempo, sus tareas se reasignaron a otro miembro con más tiempo libre. Esto evitó que el trabajo se detuviera.
- **Reuniones focalizadas y frecuentes:** Se decidió aumentar la frecuencia de las reuniones. Adicionalmente, se implementaron reuniones espontáneas para resolver dudas apenas surgieran, en estas solo participaban los miembros que estuvieran directamente involucrados con el problema.
- **Desafío con las tecnologías:** El equipo contaba con poca experiencia con las herramientas y tecnologías utilizadas para llevar a cabo el proyecto. Esto generó incertidumbre sobre cómo empezar y cuánto tiempo tomaría cada tarea.
- **Estrategias de mitigación:**
 - **Investigación previa:** Antes de comenzar a programar, el equipo dedicó tiempo a auto-capacitarse y realizar investigaciones técnicas.
 - **Desarrollo progresivo:** Se optó por planificar y programar las funcionalidades por capas en lugar de intentar crear la solución completa de inmediato. Por ejemplo, en el sistema de mensajería, primero se generaron las bases para enviar un mensaje de prueba simple al presionar el botón correspondiente (Sprint 2); posteriormente se integró el autorrellenado de datos para la plantilla (Sprint 3) y finalmente se integraron las demás plantillas para cubrir todos los casos.
- **Desafíos con el patrocinador y el entorno:** En ocasiones, el patrocinador se ausentó a diversas ceremonias, lo que dificultó la obtención de feedback. Además, al inicio existía un desconocimiento sobre la infraestructura del hospital, lo que generaba el riesgo de desarrollar una solución incompatible.
- **Estrategia de mitigación:**
 - **Reuniones preparadas:** Para aprovechar el poco tiempo del patrocinador, se coordinaron reuniones extraoficiales llevando listas de preguntas preparadas con antelación para ser más eficientes. Esto permitió resolver dudas técnicas específicas y obtener información relevante sobre la infraestructura sin depender exclusivamente de las reuniones formales.

9 Estimación de costos

El análisis económico permite dimensionar la inversión total necesaria para la materialización del proyecto. La estimación se divide en inversión inicial y costos operativos, proporcionando una visión clara de la viabilidad financiera del Producto Mínimo Viable (MVP).

9.1 Estimación de costos de recursos humanos

La valorización del capital humano se calculó utilizando un modelo de Costo por Hora Hombre (HH) en valores brutos (Costo Empresa), asumiendo una jornada estándar de 160 horas mensuales. Las tarifas referenciales fueron extraídas de los portales especializados Indeed Chile y DespegaPyme para perfiles *Junior* durante el segundo semestre de 2025. Finalmente, la distribución de horas totales de la Tabla 5 refleja la

capacidad operativa real del equipo, la cual fue ajustada según las limitaciones de disponibilidad por cargas académicas y prácticas profesionales simultáneas.

Costos de recursos humanos				
Roles	Referencia mensual	Valor hora (CLP)	Horas totales	Total estimado (CLP)
Product Owner	\$1.656.000	\$10.350	240	\$2.484.000
Scrum Master	\$1.677.440	\$10.484	360	\$3.774.240
Desarrollador Backend Junior	\$1.279.360	\$7.996	320	\$2.558.720
Desarrollador Frontend React Junior	\$1.104.000	\$6.900	250	\$1.725.000
Diseñador UI/UX Junior	\$1.476.960	\$9.231	300	\$2.769.300
Total			1.470	\$13.311.260

Tabla 5. Costos de recursos humanos

9.2 Costos operativos

A diferencia de la inversión inicial, los costos operativos representan el gasto necesario para mantener el sistema en funcionamiento de forma continua. Al ser un sistema basado en la API de WhatsApp Cloud, el costo es proporcional al uso.

Costos operativos				
Ítem	Detalle	Costo unitario	Cantidad	Total mensual
WhatsApp Cloud API	Categoría "Utilidad"	\$0.02 USD/Mensaje	1800 mensajes	\$36 (USD)
Dominio/SSL	Certificado para el Webhook	\$10.000/año	1/12	\$834 (CLP)
Electricidad + internet	Consumo del equipo	Estimado	1	\$15.000 (CLP)
			Total mensual	~\$49.040 (CLP)

Tabla 6. Costos operativos

9.3 Costos de infraestructura y hardware

Corresponde a una inversión única en activos físicos necesarios para el despliegue local en el hospital.

Costos de infraestructura		
Ítem	Detalle	Total estimado (CLP)
Servidor On-Premise	Pc/Notebook/All in One	\$700.000
Infraestructura de Red	Configuración de Router/Firewall	\$50.000
	Total infraestructura	\$750.000

Tabla 7. Costos de infraestructura

9.4 Costos de licencias de herramientas

Costos en herramientas para la gestión, diseño y control de versiones durante los cuatro meses de desarrollo.

Costos de herramientas				
Herramienta	Plan	Costo unitario (mensual)	Cálculo (5 usuarios x 4 meses)	Total estimado (CLP)
Jira Software	Standard	\$9.05 (USD)	5 us. x \$9.05 x 4 meses	~\$167.000
Confluence	Standard	\$6.70 (USD)	5 us. x \$6.70 x 4 meses	~\$123.600
Figma	Professional (Full Seat)	\$20 (USD)	1 us. x \$20 x 4 meses	~\$73.800
Figma	Professional	\$15 (USD)	1 us. x \$15 x 4	~\$55.400

	(Dev Seat)		meses	
Figma	Professional (Collab Seat)	\$5 (USD)	3 us. x \$5 x 4 meses	~\$55.400
GitHub	Team	\$4 (USD)	5 us. x \$4 x 4 meses	~\$73.800
			Total Herramientas	~\$549.000

Tabla 8. Costos de herramientas

9.5 Resumen de inversión total

La inversión total para la ejecución y puesta en marcha del MVP se resume en la siguiente tabla, consolidando los montos de capital humano, infraestructura y herramientas:

Valoración de inversión	
Ítem	Valor total (CLP)
Recursos Humanos	\$13.311.260
Infraestructura	\$750.000
Licencias de Software	\$549.000
Costos durante el desarrollo	\$63.400
Total Inversión	\$14.673.660

Tabla 9. Valoración de la inversión

9.6 Conclusión económica

Se concluye que el proyecto presenta una alta viabilidad financiera desde la perspectiva de la gestión hospitalaria pública. Aunque la inversión inicial bordea los 12 millones de pesos, el costo de mantenimiento mensual es extremadamente bajo en comparación con el impacto económico del ausentismo.

La rentabilidad de la inversión (ROI) no se mide únicamente en términos de ingreso directo, sino en el ahorro de recursos críticos. Al optimizar la asistencia en la Unidad de Endoscopia, se asegura que las horas de pabellón e insumos médicos de alto costo sean utilizados eficientemente. El retorno de la inversión se alcanza al evitar la pérdida de productividad de los equipos médicos y al reducir las listas de espera mediante una agenda médica más dinámica y confirmada.

10 Conclusión

El desarrollo del presente trabajo de titulación culminó con la construcción de un Producto Mínimo Viable (MVP) enfocado en la Unidad de Endoscopía del Hospital Dr. Gustavo Fricke. Al finalizar los ciclos de desarrollo, se concluye que los objetivos específicos propuestos para este proyecto fueron cumplidos en su totalidad, logrando validar una solución tecnológica efectiva frente a la problemática del ausentismo de pacientes (*No-Show*).

A continuación, se detalla cómo se logró el cumplimiento de cada uno de los objetivos específicos planteados:

- **Gestión de tiempos para generar incrementos de valor:** Este objetivo se abordó utilizando un cronograma integrado con un tablero visual Scrum. En este contexto, las fechas límite no funcionaron como topes rígidos, sino más bien como indicadores o alertas tempranas. Al visualizar que una tarea corría el riesgo de atrasarse frente a su fecha límite, esta alerta actuaba como un "interruptor" que activaba la redistribución del trabajo entre los integrantes, asegurando así la entrega a tiempo de los incrementos funcionales al cierre de cada *Sprint*.
- **Adaptación del marco Scrum al contexto estudiantil:** Se adaptaron las ceremonias ágiles para no entorpecer los horarios de clases y trabajo. En lugar de las tradicionales reuniones diarias (*Daily Scrums*), el equipo se organizó mediante tres o cuatro sesiones semanales de sincronización, sumando reuniones breves y espontáneas únicamente cuando surgían dudas o problemas técnicos que requerían atención urgente.
- **Planificación y asignación de tareas según cargas académicas:** La distribución del trabajo se ejecutó aplicando una gestión de capacidad asimétrica. Se tomó en consideración que dos integrantes del equipo se encontraban cursando su práctica profesional, por lo cual se les asignó un volumen menor de historias de usuario y tareas de menor complejidad técnica, garantizando que su disponibilidad reducida no se convirtiera en un cuello de botella para el proyecto.
- **Supervisión del progreso y detección de mejoras:** El monitoreo constante del proyecto permitió identificar desviaciones a tiempo. Si bien en etapas tempranas se detectaron deficiencias de comunicación que obligaron a rehacer ciertas funcionalidades, estas situaciones fueron tratadas como oportunidades de mejora. A medida que el equipo dominó las tecnologías utilizadas, fue posible optimizar el desarrollo e implementar características de valor agregado no contempladas en el diseño inicial del sistema.
- **Uso de herramientas ágiles para la resolución de problemas:** La integración de herramientas de gestión de proyectos y canales de comunicación instantánea (como Discord y repositorios colaborativos) centralizó el flujo de información. Esto permitió que cualquier impedimento o bloqueo fuera visibilizado al instante, facilitando que el equipo interviniera y apoyara en la resolución de errores de código o diseño de manera colaborativa, evitando el aislamiento de las tareas.

En resumen, la construcción de este MVP fue mucho más que solo programar; se convirtió en un ejercicio práctico sobre cómo enfrentar problemas reales. De esta experiencia destacan tres aprendizajes principales:

1. La importancia de la capacidad de adaptación: Usar metodologías ágiles fue fundamental. Ver los cambios como algo positivo permitió que el sistema siempre respondiera a lo que el hospital realmente necesitaba. Si se hubiera usado un enfoque más rígido y tradicional, el riesgo de fracaso habría sido mucho mayor.

2. La superación de desafíos logísticos grupales: Organizar el trabajo con disponibilidades tan limitadas requirió mucha madurez. El proyecto salió adelante gracias a la buena comunicación y a la disposición para repartir la carga cuando era necesario, demostrando que el grupo logró consolidarse como un verdadero equipo frente a las dificultades.

3. La evolución técnica y el aprendizaje: El crecimiento formativo fue enorme. Se pasó de un desconocimiento inicial sobre las tecnologías a la construcción de una plataforma completa y escalable. Todo este avance técnico quedó validado con los excelentes comentarios que recibió el proyecto durante su presentación en la Feria de Software (FESW).

Proyección y Trabajo Futuro

Actualmente, el sistema se encuentra en un estado de Producto Mínimo Viable (MVP) completamente funcional en un entorno de pruebas. El próximo paso lógico para la institución, previo a cualquier intento de implementación oficial en su red, será someter el sistema a auditorías de ciberseguridad y aplicar protocolos avanzados de encriptación y anonimización de datos.

Una vez superada esta barrera técnica y legal respecto al manejo de información sensible, la arquitectura modular desarrollada permitirá que el sistema no solo opere de manera oficial en la Unidad de Endoscopia, sino que cuente con la capacidad de escalar hacia cualquier otro servicio clínico del recinto hospitalario.

Finalmente, se identifica que el potencial de la herramienta trasciende el ámbito de la salud pública. Debido a su diseño paramétrico, la solución posee la versatilidad necesaria para adaptarse a otros sectores económicos que dependan de la gestión de citas y agendas, tales como clínicas dentales, centros de estética, consultorías especializadas o cualquier industria de servicios que requiera optimizar su tasa de asistencia mediante una comunicación automatizada y directa con el usuario. Esta capacidad de transpolar la lógica del sistema a distintos rubros comerciales posiciona al proyecto como una base sólida para futuros emprendimientos o soluciones de software como servicio (SaaS).

Referencias

- DespegaPyme. (2025). *Mejores trabajos tech 2025 en Chile: sueldos, requerimientos y proyecciones*. Recuperado de [Mejores trabajos tech 2025 en Chile: sueldos, requerimientos y proyecciones – DespegaPyme](#)
- Hiya. (2024). *Global Call Threat Report Q4 2024*. [Global Call Threat Report Q4 2024](#)
- Hiya. (2025). *Global Call Threat Report Q1 2025*. [Hiya Informe global sobre amenazas de llamadas, primer trimestre de 2025](#)
- Indeed. (2025). *Sueldo base promedio de Desarrollador backend en Chile (Reporte histórico 2025)*. Recuperado de <https://cl.indeed.com/career/desarrollador-backend/salaries>
- Indeed. (2025). *Sueldo base promedio de Desarrollador front end en Chile (Reporte histórico 2025)*. Recuperado de <https://cl.indeed.com/career/desarrollador-front-end/salaries>
- Indeed. (2025). *Sueldo base promedio de Diseñador/a ux en Chile (Reporte histórico 2025)*. Recuperado de <https://cl.indeed.com/career/dise%C3%B1ador-ux/salaries>
- Indeed. (2025). *Sueldo base promedio de Product Owner en Chile (Reporte histórico 2025)*. Recuperado de <https://cl.indeed.com/career/product-owner/salaries>
- Indeed. (2025). *Sueldo base promedio de Scrum Master en Chile (Reporte histórico 2025)*. Recuperado de <https://cl.indeed.com/career/scrum-master/salaries>

Bibliografía

- Agile Alliance. (2001). *Manifiesto for Agile Software Development*. Recuperado de <https://agilemanifesto.org/>
- Atlassian. (s.f.). *Los tres pilares del scrum: conoce los principios fundamentales del scrum*. Recuperado de <https://www.atlassian.com/es/agile/project-management/3-pillars-scrum>
- Atlassian. (s.f.). *Qué es scrum y cómo empezar*. Recuperado de <https://www.atlassian.com/es/agile/scrum>
- Atlassian. (s.f.). *Tablero de scrum*. Recuperado de <https://www.atlassian.com/es/agile/project-management/scrum-board>
- Lark Suite. (s.f.). *Tablero Scrum vs Tablero Kanban: Diferencias clave*. Recuperado de https://www.larksuite.com/es_mx/blog/scrum-board-vs-kanban-board
- Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum Guides. Recuperado de <https://scrumguides.org/scrum-guide.html>