

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
SEDE VIÑA DEL MAR – JOSÉ MIGUEL CARRERA

PLATAFORMA WEB DE REPARACIONES DOMICILIARIAS

Trabajo de título para optar al título de
Ingeniero de Ejecución en Software.

Alumno:

Matías Andrés Leal Rivera

Profesor guía:

Óscar Francisco Carrasco Vera



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: Plataforma Web De Reparaciones Domiciliarias

Nombre del candidato(a): Matías Andrés Leal Rivera

Carrera / Grado: Ingeniería de Ejecución de Software

Campus: Viña del Mar ; **Departamento:** Electrotecnia e Informática

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Óscar Carrasco Vera, 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

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

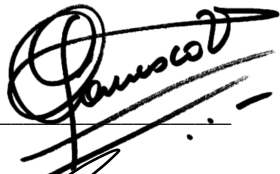
☐ El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (embargo) por:

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

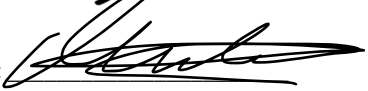
Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

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

Fecha: 8/8/2025; **Firma:** 

Estudiante o Candidato(a):

Fecha: 8/8/2025; **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.

Doy gracias a la vida por esta oportunidad de cumplir con esta meta.

RESUMEN

KEYWORDS: REPARACIONES DOMICILIARIAS, MODELO DE COTIZACIÓN MÚLTIPLE, CALIFICACIONES Y RESEÑAS

Las aplicaciones y plataformas web han revolucionado la forma de vivir, trabajar, comunicarse y resolver problemáticas, en la actualidad las personas puedes pedir servicios de transporte, alimentos o encomiendas, pero cuando se habla de servicios de reparación y mantenimiento del hogar, las aplicaciones y plataformas web se encuentran al debe.

En general, cuando un propietario busca solucionar sus problemas en el hogar, suele recurrir al primer servicio que encuentre o que se le recomienda, sin analizar si esta es la mejor opción. Por lo cual, se busca desarrollar un sistema, el cual pueda gestionar esta problemática y entregarle al propietario un desglose de opciones describiendo la solución que ofrecen profesionales, con sus costos y referencias, dadas por otros usuarios, de esta manera los clientes puedes ser conscientes de sus opciones.

Este sistema se busca desarrollar de forma interna como empresa, lo cual se decide bajo la necesidad de una mayor personalización, escalabilidad y funcionalidades. Esta opción permite alinear completamente el sistema con los objetivos específicos del proyecto y las expectativas de los usuarios. En comparación de un desarrollo externo o la compra de un software ya desarrollado, aunque ambas opciones sean más rápidas de implementar, no logra el cometido debido a sus limitaciones en términos de flexibilidad, control y adaptación a largo plazo.

Por otro lado, el sistema presenta una matriz de trazabilidad que inicia con el registro y autenticación de los usuarios, pasa por casos como evaluación y propuesta de profesionales, programación y gestión de visitas, y feedback y calificaciones, finalizando con el historial de servicios. En donde cada caso busca que el cliente pueda elegir su mejor opción en conocimiento de costos y opciones de otros usuarios.

Dado que se centró en un desarrollo propio, se diseñó la arquitectura del sistema considerando aspectos clave como la base de datos, la interfaz de usuario y los modelos de seguridad, con el propósito de garantizar un funcionamiento eficiente, seguro y escalable.

ÍNDICE

INTRODUCCIÓN 1

1. ASPECTOS RELEVANTES DEL PROYECTO Y SU GESTIÓN..... 3

1.1. Descripción de la organización 3

1.2. Descripción de la situación sin proyecto 5

1.3. Problemas detectados 6

1.4. Objetivos (General y Específicos)..... 7

1.5. Requerimientos del usuario 8

1.6. Alternativas globales de solución..... 9

1.6.1. Alternativa 1: Desarrollo Propio..... 9

1.6.2. Alternativa 2: Externalizar el Desarrollo 10

1.6.3. Alternativa 3: Comprar software ya desarrollado.....10

1.7. Criterios para evaluar las alternativas.....11

1.7.1. Criterios para evaluar alternativas. 11

1.7.2. Ponderación de Criterios de Evaluación y Escala de Evaluación 12

1.7.3. Evaluación.....13

1.8. Alternativa seleccionada 15

1.8.1. Descripción.....15

1.8.2. Objetivo de la alternativa seleccionada.....16

1.8.3. Beneficios.....16

1.8.4. Desarrollo del plan de personal 17

1.8.5. Planificación temporal, y asignación de recursos (Carta Gantt) 18

1.8.6. Administración del riesgo 20

1.8.7. Estimación de costos.....21

2. ASPECTOS RELEVANTES DEL ANÁLISIS. 23

2.1. Descripción de la solución propuesta 23

2.2. Diagrama de actividades general UML (flujo administrativo) 24

2.1 Requerimientos del sistema 26

2.3.1 Requerimientos funcionales 27

2.3.2 Requerimientos no funcionales 28

2.2 Estructura funcional.....29

2.3 Matriz de trazabilidad 31

2.4 Modelo Conceptual 32

2.5 Modelo de Casos de Uso 34

2.5.1 Caso de uso: CU01- Crear Solicitud.....34

2.5.2 Caso de uso: CU02 - Publicar Solicitud.....36

2.5.3	Caso de uso: CU03 - Cancelar Solicitud.....	38
2.5.4	Caso de uso: CU04 - Seleccionar Propuestas.....	40
2.5.5	Caso de uso: CU05 - Acceder Historial.....	42
2.5.6	Caso de uso: CU06 - Gestionar Perfil.....	44
2.5.7	Caso de uso: CU07 - Generar Evaluación.....	46
2.5.8	Caso de uso: CU08 - Finalizar Servicio.....	48
2.5.9	Caso de uso: CU09 – Procesar Servicio.....	50
2.5.10	Caso de uso: CU10 – Crear Servicio.....	52
2.5.11	Caso de uso: CU11 – Gestionar Servicio.....	54
2.5.12	Caso de uso: CU12 – Administrar Notificaciones.....	56
2.5.13	Caso de uso: CU13 – Administrar Usuarios.....	58
2.5.14	Caso de uso: CU14 – Gestionar Conflictos.....	60
2.5.15	Caso de uso: CU15 – Administrar Solicitudes.....	62
2.5.16	Caso de uso: CU16 – Generar Reporte.....	64
3.	ASPECTOS RELEVANTES DEL DISEÑO.....	67
3.1.	Arquitectura del software.....	67
3.1.1.	Arquitectura del sistema.....	67
3.1.2.	Características del recurso computacional.....	68
3.2.	Diseño de datos.....	69
3.2.1.	Modelo de clases.....	69
3.2.2.	Modelo relacional.....	71
3.2.3.	Diccionario de datos.....	73
3.3.	Diagrama de secuencia extendido.....	78
3.4.	Diseño de interfaz.....	91
3.4.1.	Inicio cliente.....	91
3.4.2.	Mi perfil – Cliente.....	92
3.4.3.	Ingresar solicitud servicio – Cliente.....	93
3.4.4.	Ver solicitudes – Cliente.....	94
3.4.5.	Ver solicitudes – Profesional.....	95
3.4.6.	Ver detalle solicitud – Profesional.....	96
3.4.7.	Inicio administrador.....	97
	CONCLUSION	98
	BIBLIOGRAFÍA	100

ÍNDICE DE FIGURAS

Figura 2.1. Diagrama de Actividad UML del Flujo de Solicitud de Servicio 25

Figura 2.2 Caso de uso general.....29

Figura 2.3. Modelo conceptual.....33

Figura 2.4. Diagrama de secuencia: CU01- Crear Solicitud 35

Figura 2.5. Diagrama de secuencia 02 - Publicar Solicitud 37

Figura 2.6. Diagrama de secuencia: CU03 - Cancelar Solicitud 39

Figura 2.7. Diagrama de secuencia 04 - Seleccionar Propuestas 41

Figura 2.8. Diagrama de secuencia: CU05 - Acceder Historial 43

Figura 2.9. Diagrama de secuencia: CU06 - Gestionar Perfil 45

Figura 2.10. Diagrama de secuencia: CU07 - Generar Evaluación.....47

Figura 2.11. Diagrama de secuencia: CU08 - Finalizar Servicio 49

Figura 2.12. Diagrama de secuencia: CU09 – Procesar Servicio 51

Figura 2.13. Diagrama de secuencia: CU10 – Crear Servicio 53

Figura 2.14. Diagrama de secuencia: CU11 – Gestionar Servicio 55

Figura 2.15. Diagrama de secuencia: CU12 – Administrar Notificaciones 57

Figura 2.16. Diagrama de secuencia: CU13 – Administrar Usuarios.....59

Figura 2.17. Diagrama de secuencia: CU14 – Gestionar Conflictos.....61

Figura 2.18. Diagrama de secuencia: CU15 – Administrar Solicitudes 63

Figura 2.19. Diagrama de secuencia: CU16 – Generar Reporte 65

Figura 3.1. Arquitectura del sistema 68

Figura 3.2. Modelo de clases 70

Figura 3.3. Modelo Relacional 72

Figura 3.4. Diagrama de secuencia extendido: CU01 – Crear Solicitud 78

Figura 3.5. Diagrama de secuencia extendido: CU02 – Publicar Solicitud 78

Figura 3.6. Diagrama de secuencia extendido: CU03 Cancelar Solicitud 79

Figura 3.7. Diagrama de secuencia extendido: CU04 – Seleccionar Propuesta.....79

Figura 3.8. Diagrama de secuencia extendido: CU05 – Acceder Historial 80

Figura 3.9. Diagrama de secuencia extendido: CU06 – Gestionar Perfil 80

Figura 3.10. Diagrama de secuencia extendido: CU07 – Generar Evaluación 81

Figura 3.11. Diagrama de secuencia extendido: CU08 – Finalizar Servicio 82

Figura 3.12. Diagrama de secuencia extendido: CU09 – Procesar Servicio 83

Figura 3.13. Diagrama de secuencia extendido: CU10 – Crear Servicio.....84

Figura 3.14. Diagrama de secuencia extendido: CU11 – Gestionar Servicio 85

Figura 3.15. Diagrama de secuencia extendido: CU12 – Administrar Notificaciones 86

Figura 3.16. Diagrama de secuencia extendido: CU13 – Administrar Usuarios.....87

Figura 3.17. Diagrama de secuencia extendido: CU14 – Gestionar Conflictos.....88

Figura 3.18. Diagrama de secuencia extendido: CU15 – Administrar Solicitudes 89

Figura 3.19. Diagrama de secuencia extendido: CU16 – Generar Reporte 90

Figura 3.20. Interfaz Inicio Cliente.....91

Figura 3.21. Interfaz Mi Perfil Cliente.....92

Figura 3.22. Interfaz Formulario Solicitud.....93

Figura 3.23. Interfaz Ver Solicitud Cliente 94

Figura 3.24. Ver Solicitudes Profesional 95

Figura 3.25. Interfaz Ver Detalle Solicitud Profesional.....96

Figura 3.26. Interfaz Inicio Administrador 97

ÍNDICE DE TABLAS

Tabla 1.1. Requerimientos del usuario 8

Tabla 1.2. Ponderación de los criterios de evaluación 12

Tabla 1.3. Escala de Calificación.....12

Tabla 1.4. Evaluación: Alternativa 1 - Desarrollo Propio 13

Tabla 1.5. Evaluación: Alternativa 2 - Externalizar el Desarrollo.....14

Tabla 1.6. Evaluación: Alternativa 3 - Comprar software ya desarrollado 14

Tabla 1.7. Resumen Evaluación de alternativas de solución.....15

Tabla 1.8. Programación temporal del proyecto.....19

Tabla 1.9. Evaluación de Riesgos 20

Tabla 1.10. Plan de mitigación 20

Tabla 1.11. Estimación de costos por recurso 21

Tabla 2.1. Requerimientos Funcional 27

Tabla 2.2. Requerimientos No Funcionales 28

Tabla 2.3. Matriz de trazabilidad del sistema 31

Tabla 2.4. Caso de uso narrativo: CU01- Crear Solicitud.....34

Tabla 2.5. Contrato: CU01- Crear Solicitud.....35

Tabla 2.6. Caso de uso narrativo: CU02 - Publicar Solicitud 36

Tabla 2.7. Contrato: CU02 - Publicar Solicitud 37

Tabla 2.8. Caso de uso narrativo: CU03 - Cancelar Solicitud.....38

Tabla 2.9. Contrato: CU03 - Cancelar Solicitud.....39

Tabla 2.10. Caso de uso narrativo: CU04 - Seleccionar Propuestas.....40

Tabla 2.11. Contrato 04 - Seleccionar Propuestas 41

Tabla 2.12. Caso de uso narrativo 05 Acceder Historial.....42

Tabla 2.13. Contrato: CU05 - Acceder Historial 43

Tabla 2.14. Caso de uso narrativo: CU06 - Gestionar Perfil.....44

Tabla 2.15. Contrato: CU06 - Gestionar Perfil.....45

Tabla 2.16. Caso de uso narrativo: CU07 - Generar Evaluación.....46

Tabla 2.17. Contrato: CU07 - Generar Evaluación.....47

Tabla 2.18. Caso de uso narrativo: CU08 - Finalizar Servicio 48

Tabla 2.19. Contrato: CU08 - Finalizar Servicio 49

Tabla 2.20. Caso de uso narrativo: CU09 – Procesar Servicio.....50

Tabla 2.21. Contrato: CU09 – Procesar Servicio.....51

Tabla 2.22. Caso de uso narrativo: CU10 – Crear Servicio 52

Tabla 2.23. Contrato: CU10 – Crear Servicio 53

Tabla 2.24. Caso de uso narrativo: CU11 – Gestionar Servicio 54

Tabla 2.25. Contrato: CU11 – Gestionar Servicio 55

Tabla 2.26. Caso de uso narrativo: CU12 – Administrar Notificaciones.....56

Tabla 2.27. Contrato: CU12 – Administrar Notificaciones.....57

Tabla 2.28. Caso de uso narrativo: CU13 – Administrar Usuarios 58

Tabla 2.29. Contrato: CU13 – Administrar Usuarios 59

Tabla 2.30. Caso de uso narrativo: CU14 – Gestionar Conflictos.....60

Tabla 2.31. Contrato: CU14 – Gestionar Conflictos.....61

Tabla 2.32. Caso de uso narrativo: CU15 – Administrar Solicitudes 62

Tabla 2.33. Contrato: CU15 – Administrar Solicitudes 63

Tabla 2.34. Caso de uso narrativo: CU16 – Generar Reporte 64

Tabla 2.35. Contrato: CU16 – Generar Reporte 65

Tabla 3.1. Diccionario de datos: Administrador.....73

Tabla 3.2. Diccionario de datos tabla: Cliente.....73

Tabla 3.3. Diccionario de datos tabla: Solicitud.....74

Tabla 3.4. Diccionario de datos tabla: Servicio 74

Tabla 3.5. Diccionario de datos tabla: Profesional 75

Tabla 3.6. Diccionario de datos tabla: Evaluación 75

Tabla 3.7. Diccionario de datos tabla: Conflicto 76

Tabla 3.8. Diccionario de datos tabla: Propuesta.....76

Tabla 3.9. Diccionario de datos tabla: Notificacion 77

INTRODUCCIÓN

En la actualidad, los problemas en el hogar, tales como, fugas de agua, cortocircuitos eléctricos y otros problemas de mantenimiento suelen interrumpir la rutina y causar estrés en los propietarios. Además, encontrar profesionales confiables para resolver estos problemas puede ser un proceso complicado y confuso, dado que, frente a la urgencia suelen contactar a cualquier persona para solucionar el problema, sin evaluar si esta es su mejor opción.

Para resolver esta problemática, se busca diseñar e implementar un sistema en donde gestionar reparaciones y mantenimientos en el hogar sea tan sencillo como solicitar un servicio a través de una plataforma web. Los usuarios pueden presentar sus requerimientos mediante un formulario sencillo; esta solicitud es revisada por una red de maestros especializados que evaluarán la situación y proporcionarán un costo estimado para la solución. De esta forma, los usuarios tienen a su disposición una amplia selección de profesionales calificados, los cuales son evaluados y calificados por otros usuarios dentro de la misma plataforma. Los propietarios pueden comparar tarifas y calificaciones, seleccionar al experto que mejor se adapte a sus necesidades, y programar el servicio de manera conveniente y totalmente en línea.

Finalmente, esta solución incluye una transparencia de costos, del mismo modo que se utiliza en las aplicaciones para solicitar servicios de transporte, en donde, se tiene conocimiento del costo antes de abordar el vehículo. Los propietarios pueden tener una estimación clara de los costos que traerá la reparación o mantenimiento de sus hogares antes de comprometerse con el servicio.

CAPÍTULO 1. ASPECTOS RELEVANTES DEL PROYECTO Y SU GESTIÓN

1. ASPECTOS RELEVANTES DEL PROYECTO Y SU GESTIÓN

La Plataforma Web de Reparaciones Domiciliarias basado en el modelo de cotización múltiple busca transformar la forma en que los propietarios manejan las reparaciones y el mantenimiento en sus hogares. Al brindar conveniencia, transparencia y la posibilidad de elegir entre profesionales de confianza. Esta plataforma busca aliviar la carga de las tareas de mantenimiento y permitir que los propietarios cuiden de sus hogares de manera eficiente.

1.1. Descripción de la organización

Triangle Tech, ubicada en Hendaya 60, piso 7, Las Condes, Región Metropolitana, y fundada en el año 2018, fue presentada como una empresa dedicada a ofrecer soluciones informáticas.

Se estableció que su misión consiste en la provisión de soluciones tecnológicas avanzadas y personalizadas, orientadas a satisfacer las necesidades específicas de los clientes y contribuir de manera significativa a su éxito y crecimiento sostenible. Esta visión se fundamentó en la tecnología como un medio poderoso para fomentar la innovación y optimizar los procesos empresariales.

El enfoque de la empresa se estructuró en torno a cinco pilares clave:

- **Innovación Tecnológica:** La empresa se propuso mantenerse a la vanguardia de las tendencias tecnológicas, aplicando avances innovadores para resolver desafíos empresariales y generar nuevas oportunidades.
- **Calidad y Excelencia:** Se definió la calidad como un principio rector, esforzándose por ofrecer productos y servicios de alta calidad, respaldados por estándares rigurosos de control y pruebas exhaustivas.
- **Colaboración:** Se determinó que la colaboración resultaba clave para el éxito, tanto interno como externo, destacando el trabajo conjunto con clientes, asociados y equipos internos para fomentar la innovación y el logro compartido.
- **Ética y Sostenibilidad:** Se definió la importancia de operar bajo principios éticos y sostenibles, reafirmando un compromiso con prácticas comerciales responsables que beneficiaran a la sociedad y al medio ambiente.
- **Desarrollo de la Comunidad:** Se afirmó que una empresa exitosa debía contribuir al desarrollo de la comunidad en la que operaba, apoyando iniciativas locales y proyectos que contienen un impacto positivo en la sociedad.

Se concluyó que la misión de Triangle Tech iba más allá de la provisión de servicios tecnológicos, estableciendo un compromiso profundo con el crecimiento y la eficiencia de sus clientes. Se destacó que la empresa promovía un enfoque ético y sostenible en todas sus actividades, buscando posicionarse como una empresa confiable y valiosa para las organizaciones interesadas en maximizar el potencial de la tecnología.

Se definió que la visión de Triangle Tech es liderar el mercado de soluciones informáticas en Chile, destacándose por la excelencia en el servicio al cliente, la innovación tecnológica y el compromiso con la comunidad. Para alcanzar esta visión, se propusieron los siguientes objetivos:

- Desarrollar soluciones tecnológicas a medida que satisfagan las necesidades específicas de los clientes.
- Mantener una posición de vanguardia de las últimas tendencias tecnológicas para ofrecer productos y servicios de alta calidad.
- Fomentar la colaboración interna y externa para impulsar la innovación y el crecimiento continuo.
- Contribuir al desarrollo sostenible de la sociedad a través de prácticas comerciales éticas y sostenibles.

Triangle Tech ofrece productos y servicios de informática, entre los que se incluían:

- Desarrollo de software personalizado para empresas.
- Consultoría en tecnologías de la información.
- Servicios de ciberseguridad y protección de datos.
- Gestión de proyectos tecnológicos.
- Implementación de soluciones de inteligencia artificial y análisis de datos.

Los departamentos que participarían en el desarrollo del sistema, asignando las siguientes responsabilidades:

- Desarrollo de Software: Diseño y creación del sistema.
- Consultoría: Asesoramiento para la integración del sistema en las operaciones de los clientes.
- Ventas y Marketing: Promoción y venta del sistema a los clientes.
- Soporte Técnico: Provisión de asistencia continua a los usuarios del sistema.

1.2. Descripción de la situación sin proyecto

Se identificó que cuando una persona enfrentaba inconvenientes en el hogar, resultaba complejo encontrar al profesional adecuado para resolver dichas problemáticas. Por lo general, se recurre a recomendaciones de familiares, búsquedas en internet u otras alternativas que no garantiza la solución del problema. En ciertos casos, esta situación incluso derivaba en un empeoramiento de la problemática inicial.

De acuerdo con los datos del CENSO 2017, se registró que, para esa fecha, la población en Chile alcanzaba los 17.574.003 habitantes, cifra que ha aumentado en los últimos años. Este sondeo reveló que el 65% de los chilenos poseía una vivienda propia, ya sea completamente pagada o en proceso de pago, mientras que el 25% arrendaba una vivienda. Esto implica que una parte significativa de la población enfrenta la necesidad de solicitar ayuda para solucionar inconvenientes en el hogar. Sin embargo, se identificó que la falta de una plataforma confiable y segura conducía a trabajos mal realizados, falta de profesionalismo, precios elevados y, sobre todo, a una ausencia de confianza e incertidumbre, dado que la mayoría de las personas suele contratar a desconocidos sin verificar sus antecedentes o calificaciones.

Se señaló que los propietarios enfrentan inseguridades y riesgos al contratar a profesionales sin una adecuada verificación, lo que derivaba en reparaciones de baja calidad y costos más altos a largo plazo. Además, se señaló que la ausencia de un sistema que visibilizara la experiencia de los profesionales y permite mostrar retroalimentación de los clientes sobre los trabajos realizados, lo que garantizaría la calidad del servicio. La falta de un mecanismo centralizado de comunicación y coordinación entre propietarios y profesionales también complicaba el proceso, generando malentendidos en relación con costos, tiempos de ejecución y alcance de los trabajos. Esto, a su vez, incrementa el estrés y el tiempo invertido en la gestión de reparaciones.

Por otra parte, se indicó que los profesionales del sector de reparaciones también enfrentan dificultades para captar clientes, dependiendo en gran medida de recomendaciones personales o medios poco confiables. Esta situación limita su visibilidad y oportunidades laborales, al tiempo que se generaba una competencia desleal, donde profesionales capacitados se ven obligados a competir con otros menos calificados que ofrecen precios bajos, devaluando la calidad de los servicios.

Asimismo, se identificó que la falta de una plataforma organizada y confiable dificultaba la gestión de emergencias y el mantenimiento preventivo en los hogares. En situaciones de urgencia, como fugas de agua o fallas eléctricas, los propietarios experimentan largos tiempos de espera y precios elevados debido a la disponibilidad limitada de profesionales confiables. Además, la ausencia de un sistema que promoviera el mantenimiento preventivo provoca postergación de

reparaciones menores, las cuales se transforman en problemas más graves y costosos, afectando la calidad de vida de los propietarios.

En conclusión, se destaca que la situación actual, sin una Plataforma Web de Reparaciones Domiciliarias que facilitara la búsqueda, contratación y coordinación de servicios, impactaba negativamente en la calidad y el costo de las reparaciones. Además, esta carencia impedía una experiencia satisfactoria tanto para propietarios como para profesionales, dejando a ambos en una posición de incertidumbre y desconfianza.

1.3. **Problemas detectados**

Problemas asociados a la problemática que el sistema busca resolver:

1. **Dificultad para encontrar profesionales confiables:** Se observó que los propietarios enfrentaban incertidumbre al intentar localizar profesionales que realicen mantenimiento y reparación con calificaciones y experiencias verificables.
2. **Falta de transparencia en los costos:** Se señaló que la ausencia de información clara sobre los costos de reparación y mantenimiento generaba inconvenientes, como sorpresas inesperadas en la factura final.
3. **Gestión desorganizada:** Se destacó que la carencia de una herramienta centralizada para gestionar reparaciones y mantenimiento conducía a pérdidas de tiempo y a la acumulación de tareas pendientes.
4. **Emergencias no planificadas:** Se reconoció que los problemas domésticos, como fugas de agua o fallas eléctricas u otra situación, podían surgir inesperadamente, requiriendo respuestas inmediatas.
5. **Evaluación de la calidad:** Se observó que los propietarios experimentaban dificultades para evaluar la calidad del trabajo realizado por los profesionales y la satisfacción de otros clientes.
6. **Coordinación ineficiente:** Se concluyó que la coordinación de horarios entre propietarios y profesionales resultaba compleja, derivando en demoras y molestias.
7. **Mantenimiento preventivo descuidado:** Se determinó que la ausencia de un sistema organizado para el mantenimiento preventivo originaba daños costosos a largo plazo en las viviendas.
8. **Falta de opciones de elección:** Se indicó que la incapacidad de seleccionar entre varios profesionales limitaba la flexibilidad y las alternativas disponibles para los propietarios.

1.4. Objetivos (General y Específicos)

Se establece como objetivo general del proyecto “Diseñar y desarrollar una plataforma web orientada a optimizar y simplificar el proceso completo de gestión de reparaciones y mantenimientos domiciliarios, asegurando confianza y permitiendo a los usuarios acceder a múltiples alternativas según sus necesidades.”. Al tratarse de una plataforma web, se indica que esta permite un acceso ágil y eficiente para los usuarios desde cualquier dispositivo con conexión a internet.

Por otro lado, se definieron los siguientes objetivos específicos:

1. Simplificar la búsqueda y acceso a profesionales calificados para reparaciones y mantenimiento en el hogar.
2. Facilitar la gestión de reparaciones y mantenimiento, haciendo que resulte más conveniente y accesible para los propietarios de viviendas.
3. Proporcionar estimaciones claras de los costos de reparación o mantenimiento antes de que los usuarios se comprometieran con el servicio.
4. Asegurar que los profesionales registrados en la plataforma sean confiables y estén respaldados por calificaciones y evaluaciones de otros usuarios.
5. Permitir que los propietarios comparen tarifas y calificaciones, con el fin de tomar decisiones informadas sobre la selección de los profesionales.
6. Reducir el tiempo y el esfuerzo que es requerido para encontrar y programar servicios de reparación y mantenimiento.

1.5. Requerimientos del usuario

Las expectativas principales para los usuarios se identifican con respecto a la Plataforma Web de Reparaciones Domiciliarias, basada en el modelo de cotización múltiple. Estas expectativas se detallan en la siguiente tabla de Requerimientos del Usuario:

Tabla 1.1. Requerimientos del usuario

ID	Descripción
RU-001	Los usuarios esperan que la plataforma sea fácil de usar y les permita solicitar servicios de reparación y mantenimiento en su hogar de manera rápida y sencilla, sin complicaciones ni trámites engorrosos.
RU-002	Los usuarios desean ver de manera clara y anticipada los costos asociados a las reparaciones o el mantenimiento que soliciten, para evitar sorpresas desagradables en la factura final.
RU-003	Los usuarios esperan poder elegir entre una variedad de profesionales de reparación y mantenimiento, evaluando sus calificaciones, tarifas y reseñas de otros usuarios para seleccionar al experto que mejor se adapte a sus necesidades.
RU-004	Los usuarios confían en que los profesionales registrados en la plataforma sean confiables, estén debidamente calificados y tengan buenas referencias de otros clientes.
RU-005	Los usuarios esperan que la plataforma les ayude a organizar y gestionar las reparaciones y el mantenimiento de su hogar de manera eficiente, permitiéndoles programar citas y seguirlas de manera efectiva.
RU-006	Los usuarios desean que la plataforma les ayude a resolver problemas urgentes en su hogar de manera rápida y efectiva, como fugas de agua o cortocircuitos eléctricos.
RU-007	Los usuarios valoran la posibilidad de programar tareas de mantenimiento preventivo para evitar problemas futuros y mantener su hogar en buen estado.
RU-008	Los usuarios esperan que los profesionales en la plataforma ofrezcan un trabajo de alta calidad y que sus servicios estén respaldados por garantías.
RU-009	Los usuarios desean que la plataforma les brinde una experiencia sin estrés al cuidar de su hogar, eliminando la incertidumbre y la preocupación asociadas a las reparaciones y el mantenimiento.
RU-010	Los usuarios desean recibir notificaciones en tiempo real sobre el estado de sus solicitudes, citas programadas, cambios de última hora o actualizaciones por parte de los profesionales, facilitando así la gestión de las reparaciones y mantenciones.
RU-011	Los usuarios desean contar con un historial detallado de todos los servicios solicitados, incluyendo facturas, recibos y registros de comunicación, lo cual facilita la gestión de gastos y el seguimiento de las reparaciones realizadas.
RU-012	Los usuarios esperan un proceso de registro y autenticación seguro, rápido y sencillo, que les permita acceder a la plataforma con opciones de recuperación de cuenta en caso de olvido de contraseñas o problemas de acceso.

Fuente: Elaboración propia

1.6. Alternativas globales de solución

Se evalúan detalladamente las alternativas de solución disponibles para implementar el sistema de manera eficiente, identificando la opción que mejor se alinea con los objetivos, recursos y capacidades del proyecto. Para ello se consideraron tres estrategias principales: desarrollo propio, externalización del desarrollo y la compra de software preexistente. La selección de la alternativa se realiza en base a ocho criterios específicos: funcionalidad, usabilidad, mantenimiento, escalabilidad, seguridad de la información, costo, puesta en marcha y estabilidad/rendimiento. Las calificaciones asignadas a cada criterio se fundamentaron en antecedentes concretos relacionados con costos estimados, tiempos de implementación, nivel de personalización y soporte técnico disponible. A continuación, se describen las ventajas y desventajas de cada alternativa.

1.6.1. Alternativa 1: Desarrollo Propio

Esta estrategia implica el diseño y desarrollo completo de la plataforma por parte de un equipo interno, abarcando desde la fase de planificación hasta la implementación final. Esto permite mantener un control total sobre las funcionalidades, el diseño y la personalización del sistema.

Pros:

- Se establece que el desarrollo propio permite la personalización completa del software, permitiendo alinear cada funcionalidad con los requerimientos específicos de los usuarios y los objetivos del proyecto.
- Se reconoce que esta estrategia ofrece mayor flexibilidad para implementar cambios, agregar nuevas características y realizar mejoras continuas, adaptándose ágilmente a las demandas del mercado.
- Se garantiza el control total sobre las funcionalidades, la arquitectura y la seguridad del sistema, asegurando una implementación acorde a los estándares del proyecto y una protección robusta de la información.

Contras:

- Se indica que esta opción requiere una inversión inicial alta en tiempo, recursos técnicos y financieros, debido a la necesidad de contratar y gestionar un equipo completo de desarrollo.
- Se identifica que el tiempo de desarrollo podría extenderse, lo que retrasaría el lanzamiento al mercado y la generación temprana de ingresos.
- Se analiza que el riesgo de retrasos en la entrega originados por problemas técnicos, falta de experiencia del equipo o cambios en los requerimientos durante el desarrollo.

1.6.2. Alternativa 2: Externalizar el Desarrollo

Esta opción consiste en contratar a una empresa externa especializada en el desarrollo de software para construir la plataforma, aprovechando la experiencia y los recursos de terceros.

Pros:

- Se destaca que la externalización facilita el acceso a conocimientos técnicos especializados, acelerando el desarrollo y el lanzamiento de la plataforma mediante el uso de metodologías ágiles y buenas prácticas.
- Se señala que con esta alternativa se reduce la carga operativa y administrativa del equipo interno, permitiendo enfocar los esfuerzos en las estrategias de negocio.
- Se asegura que la aplicación contempla tecnologías y prácticas modernas, ya usadas e implementadas por las empresas proveedora, garantizando soluciones actualizadas y eficientes.

Contras:

- Se identifica que la falta de control directo sobre el desarrollo dificulta garantizar que el producto final cumpliera con los requisitos y expectativas.
- Se identifica costos adicionales derivados de revisiones de contrato, ajustes y posibles retrabajos, afectando el presupuesto del proyecto.
- Se analiza que el riesgo en problemas de comunicación y coordinación entre la empresa contratante y el proveedor podría generar malentendidos o retrasos.

1.6.3. Alternativa 3: Comprar software ya desarrollado

Esta alternativa consiste en adquirir un software preexistente que se adapte a las necesidades del proyecto, con la posibilidad de realizar ajustes menores para su implementación.

Pros:

- Se concluye que la implementación era rápida dado que el software adquirido estaba probado y listo para su uso, permitiendo una entrada temprana al mercado.
- Se determina que la inversión inicial es menor en comparación con el desarrollo propio ya que no se requerirá la contratación de un equipo de desarrollo completo.
- Se identifica que la delegación del mantenimiento, actualizaciones y soporte técnico al proveedor facilitaba la administración del sistema.

Contras:

- Se observa que las limitaciones en la personalización del software podrían impedir que este cumpliera completamente con los requerimientos del proyecto.
- Se señala la dependencia del proveedor para actualizaciones y soporte técnico, provocando riesgos en caso de discontinuidad del software o cambios en las condiciones del servicio.
- Se analiza que los costos continuos asociados a licencias, actualizaciones y mantenimiento incrementaban los gastos a largo plazo afectando la rentabilidad.

1.7. Criterios para evaluar las alternativas

La evaluación de alternativas es considerada como una etapa fundamental dentro del proceso de toma de decisiones para la implementación del sistema. Con este propósito, se busca recopilar y analizar distintas opciones, evaluando su factibilidad para determinar cuál se alinea mejor con los objetivos del proyecto. Para llevar a cabo esta evaluación se define los siguientes aspectos:

1.7.1. Criterios para evaluar alternativas.

A continuación, se presentan los criterios para evaluar las alternativas anteriormente informadas.

1. **Funcionalidad:** Se evaluará si la alternativa cumple con las funcionalidades requeridas para garantizar un buen funcionamiento del sistema. Este criterio examina la capacidad del software para cubrir las necesidades del usuario, incluyendo características clave como la solicitud de servicios, la comparación de profesionales, la programación de visitas, y la generación de estimaciones de costos.
2. **Usabilidad:** Se analizará la facilidad de uso de la plataforma tanto para los propietarios como para los profesionales. Se evaluará la claridad de la interfaz de usuario, la usabilidad del sistema, y la eficiencia en la navegación, asegurando que cualquier persona, independientemente de su nivel técnico y edad, pueda usar la plataforma sin dificultad.
3. **Mantención:** Se evaluará la facilidad con la que la plataforma puede ser mantenida y actualizada. Un sistema que requiere menos esfuerzo y recursos para corregir errores, implementar mejoras y adaptarse a nuevas tecnologías tiene una ventaja significativa.
4. **Escalabilidad:** Se analizará la capacidad de la plataforma para crecer y adaptarse a medida que aumenta la demanda. Este criterio evalúa si la solución puede soportar un mayor número de usuarios, transacciones y nuevas funcionalidades sin comprometer su rendimiento.

- 5. **Seguridad de la Información:** Se evaluará si la alternativa cumple con las normativas de seguridad, la protección de datos y la prevención vulnerabilidades y ataques cibernéticos, asegurando la protección de los datos personales y financieros de los usuarios.
- 6. **Costo:** Se analizará el costo total del proyecto, incluyendo desarrollo, implementación y mantenimiento. Se evaluará si la alternativa se ajusta al presupuesto disponible y si ofrece una buena relación costo-beneficio en comparación con otras opciones.
- 7. **Puesta en marcha:** Se evaluará el tiempo necesario para implementar la alternativa. Se analizará la rapidez con la que se puede poner en funcionamiento la solución, considerando las fases de desarrollo, pruebas y despliegue.
- 8. **Estabilidad y Rendimiento:** Se evaluará la capacidad del sistema para operar de manera consistente y eficiente bajo diferentes cargas de trabajo. Se analizarán aspectos técnicos como la velocidad de respuesta, la capacidad de soportar múltiples usuarios simultáneos, y la robustez del sistema para manejar situaciones de alta demanda sin afectar la experiencia del usuario.

1.7.2. Ponderación de Criterios de Evaluación y Escala de Evaluación

Tomando como los criterios señalados en el punto 1.7.1, las ponderaciones son:

Tabla 1.2. Ponderación de los criterios de evaluación

CRITERIO	PONDERACIÓN
Funcionalidad	20%
Usabilidad	20%
Mantenimiento	10%
Escalabilidad	10%
Seguridad de la información	10%
Costo	10%
Puesta en marcha	10%
Estabilidad y Rendimiento	10%
TOTAL:	100%

Fuente: Elaboración propia.

La escala de calificación se basará en una puntuación del 0 al 5, donde:

Tabla 1.3. Escala de Calificación

CALIFICACIÓN	CONCEPTO
0	No Aplica
1	Muy Mal
2	Mal
3	Regular

4	Bueno
5	Muy Bueno

Fuente: Elaboración propia.

1.7.3. Evaluación

En esta sección se describe la puntuación asignada a cada alternativa propuesta en el punto 1.4. Cada alternativa recibe una calificación conforme a los criterios establecidos en la tabla 1.3, los cuales fueron ponderados según los valores especificados en la tabla 1.2. Este proceso permite calcular la nota final ponderada de cada alternativa, facilitando la comparación y selección de la opción más adecuada para el proyecto.

Tabla 1.4. Evaluación: Alternativa 1 - Desarrollo Propio

CRITERIO	CALIFICACIÓN	JUSTIFICACIÓN
Funcionalidad	5	Se garantiza el cumplimiento total de las funcionalidades requeridas al permitir un desarrollo personalizado alineado a las necesidades específicas del proyecto.
Usabilidad	5	La plataforma ofrece una interfaz totalmente diseñada según las expectativas de los usuarios, asegurando facilidad de uso y navegación intuitiva.
Mantenimiento	3	El mantenimiento presenta un desafío moderado, dado que la personalización puede requerir actualizaciones frecuentes y ajustes específicos.
Escalabilidad	4	El diseño personalizado permite escalar el sistema progresivamente, adaptándolo a un mayor número de usuarios y funcionalidades.
Seguridad de la información	4	El control total sobre el desarrollo asegura altos estándares de seguridad, reduciendo riesgos de vulnerabilidades.
Costo	2	Los altos costos iniciales se justifican por la inversión en personalización y control del proyecto.
Puesta en marcha	3	El tiempo de implementación es moderado debido a la necesidad de desarrollar desde cero, aunque permite asegurar calidad.
Estabilidad y Rendimiento	4	La estabilidad se asegura mediante el control total del sistema, garantizando un rendimiento óptimo bajo distintas cargas.

Fuente: Elaboración propia

Tabla 1.5. Evaluación: Alternativa 2 - Externalizar el Desarrollo

CRITERIO	CALIFICACIÓN	JUSTIFICACIÓN
Funcionalidad	4	El proveedor externo puede cubrir la mayoría de las funcionalidades requeridas, aunque con limitaciones en la personalización.
Usabilidad	3	La usabilidad depende del diseño provisto por la empresa contratada, lo cual puede no alinearse completamente con las expectativas del usuario.
Mantenimiento	4	El mantenimiento se facilita al delegar tareas al proveedor, aunque puede depender de los acuerdos establecidos en el contrato.
Escalabilidad	4	La escalabilidad es eficiente debido a la experiencia técnica del proveedor, aunque puede requerir costos adicionales.
Seguridad de la información	4	El proveedor ocupa medidas de seguridad estándar, aunque el control no es absoluto para el contratante.
Costo	3	Los costos son intermedios, considerando las tecnologías basadas en la experiencia técnica ofrecida por el proveedor.
Puesta en marcha	4	La puesta en marcha es rápida gracias a la experiencia del proveedor, lo que reduce considerablemente los tiempos de desarrollo.
Estabilidad y Rendimiento	4	El rendimiento es consistente gracias a las soluciones preexistentes y probadas por el proveedor.

Fuente: Elaboración propia

Tabla 1.6. Evaluación: Alternativa 3 - Comprar software ya desarrollado

CRITERIO	CALIFICACIÓN	JUSTIFICACIÓN
Funcionalidad	3	El software preexistente puede cubrir parcialmente las funcionalidades requeridas, pero con limitaciones en la personalización y funcionalidades.
Usabilidad	4	La usabilidad es buena ya que el software está diseñado para ser intuitivo y probado previamente en otros contextos.
Mantenimiento	4	El mantenimiento es eficiente debido a que el proveedor se encarga de las actualizaciones y soporte técnico.
Escalabilidad	3	La escalabilidad es limitada, ya que las adaptaciones pueden requerir costos adicionales o no ser factibles.
Seguridad de la información	4	El software cumple con estándares de seguridad preestablecidos, aunque la dependencia del proveedor limita el control.
Costo	5	El costo inicial es bajo debido a la ausencia de un desarrollo desde cero, siendo una opción económica para el proyecto.
Puesta en marcha	5	La implementación es rápida, ya que el software está listo para ser usado con mínimas configuraciones.
Estabilidad y Rendimiento	4	El sistema ofrece un rendimiento confiable, probado previamente en diferentes escenarios y adaptado a múltiples usuarios.

Fuente: Elaboración propia.

Tabla 1.7. Resumen Evaluación de alternativas de solución

		ALTERNATIVAS		
CRITERIO	PONDERACIÓN	1	2	3
Funcionalidad	20%	5	4	3
Usabilidad	20%	5	3	4
Mantenimiento	10%	3	4	4
Escalabilidad	10%	4	4	3
Seguridad de la información	10%	4	4	4
Costo	10%	2	3	5
Puesta en marcha	10%	3	4	5
Estabilidad y Rendimiento	10%	4	4	4
TOTAL:		4	3,7	3.9

Fuente: Elaboración propia.

1.8. Alternativa seleccionada

Como se comprueba en el punto anterior, la solución óptima para abordar los requerimientos del cliente es basarse en la alternativa del desarrollo propio.

1.8.1. Descripción

El desarrollo propio implica la creación completa de la plataforma desde cero por parte de un equipo interno, lo que garantiza un control absoluto sobre cada aspecto del proyecto. Esta alternativa, basada en un modelo de cotización múltiple, considera todas las fases del ciclo de vida del desarrollo de software, desde la fase de planificación, diseño, desarrollo, pruebas, hasta la implementación y mantenimiento continuo.

Al optar por el desarrollo propio, se asegura que la plataforma se diseñe específicamente para las necesidades del proyecto, permitiendo una personalización completa que responda a los requerimientos del usuario final. Además, es adaptable y modificable según las necesidades emergentes, asegurando que la plataforma evolucione de manera ágil conforme cambien las expectativas del mercado o se detecten nuevas oportunidades de mejora.

La creación de una plataforma personalizada basada en un modelo de cotización múltiple tiene varias ventajas. Los usuarios pueden recibir varias ofertas de diferentes profesionales para un

servicio, lo que les permite comparar precios, calificaciones y características de cada profesional para seleccionar la opción que mejor se ajuste a sus necesidades. La plataforma brinda flexibilidad a los propietarios en la búsqueda del servicio adecuado, sino que también ofrece a los profesionales la oportunidad de ofrecer sus servicios a más clientes, aumentando su visibilidad y competitividad.

El control total sobre la arquitectura del sistema facilita la integración con otras plataformas y tecnologías existentes dentro de la empresa, como sistemas de facturación, análisis de datos o CRM (Customer Relationship Management). Además, al desarrollar la plataforma de manera interna, se garantiza una mayor seguridad, ya que el equipo interno tiene control absoluto sobre las políticas de seguridad, el manejo de datos sensibles y las actualizaciones del sistema.

Este enfoque también permite que las decisiones de diseño y funcionalidad se alineen directamente con los objetivos estratégicos del negocio, sin tener que depender de las limitaciones de software de terceros. De este modo, la solución se adapta perfectamente al modelo de negocio, brindando a la empresa la capacidad de ajustar y mejorar la plataforma continuamente, sin estar sujeta a las condiciones o restricciones impuestas por proveedores externos.

En resumen, el desarrollo propio garantiza una plataforma que no solo cumple con los requisitos técnicos y funcionales, sino que también se adapta de manera flexible a las demandas del mercado, proporcionando a la empresa un recurso que puede crecer y evolucionar junto con el negocio. Además, al tener el control total sobre el sistema, se asegura un mayor grado de autonomía y escalabilidad a largo plazo.

1.8.2. Objetivo de la alternativa seleccionada

Diseñar e implementar una aplicación, basada en el modelo de negocios de subastas, la cual permite a los usuarios a encontrar profesionales y, tanto ofrecer un monto por sus reparaciones, como también recibir contra oferta de parte de los profesionales, a lo que facilita a los usuarios la búsqueda de un profesional a precios referentes, así como a los profesionales aceptar trabajos de manera más rápida.

1.8.3. Beneficios

Los beneficios del proyecto fueron listados de la siguiente forma:

- **Mayor Transparencia:** La plataforma facilita que los propietarios vean calificaciones, reseñas y costos anticipados, eliminando la incertidumbre al contratar servicios.
- **Ahorro de Tiempo:** Al centralizar la búsqueda y contratación de profesionales, los usuarios reducen significativamente el tiempo invertido en gestionar reparaciones.

- **Reducción de Riesgos:** La verificación de profesionales registrados disminuye los riesgos asociados con contrataciones inseguras o de baja calidad.
- **Fomento del Mantenimiento Preventivo:** La posibilidad de agendar tareas preventivas ayuda a los propietarios a evitar reparaciones costosas en el futuro.
- **Acceso a Servicios de Emergencia:** La plataforma permite la atención rápida a problemas críticos, como fugas de agua, fallas eléctricas u otras, garantizando soluciones oportunas.
- **Optimización de Recursos:** Los profesionales pueden gestionar mejor su tiempo y agenda al tener un flujo constante de solicitudes organizadas desde la plataforma.
- **Confianza del Usuario:** La retroalimentación de otros usuarios y la transparencia en el proceso generan una experiencia más confiable.
- **Competencia entre Profesionales:** Los profesionales compiten en un mercado más justo, basado en calificaciones y experiencia, mejorando la calidad del servicio.
- **Accesibilidad desde Cualquier Dispositivo:** La plataforma web permite a los usuarios gestionar servicios desde cualquier lugar y en cualquier momento.
- **Facilidad en la Gestión Financiera:** Un historial centralizado de reparaciones, facturas y pagos ayuda a los propietarios a mantener un mejor control sobre sus gastos.
- **Aumento en la Oportunidad de Trabajo:** Al promover la contratación de profesionales calificados, la plataforma apoya la economía local y fomenta el empleo.
- **Escalabilidad del Negocio:** La solución puede adaptarse fácilmente al crecimiento del mercado, agregando nuevos servicios o ampliando su alcance geográfico.
- **Mejora de la Calidad de Vida:** Los propietarios enfrentan menos estrés al saber que cuentan con un sistema confiable para cuidar su hogar.

1.8.4. Desarrollo del plan de personal

En este apartado se especifican los recursos humanos necesarios para el desarrollo de la plataforma, así como los roles y funciones que desempeñarán durante el ciclo de vida del proyecto. El equipo está conformado por los siguientes perfiles clave:

- **Jefe de Proyecto (1 persona):** Responsable de supervisar el desarrollo general del proyecto, asegurando que todas las tareas se completen en los plazos establecidos y conforme a los objetivos planteados. También se encarga de la coordinación entre los diferentes equipos y actores involucrados en el proyecto, así como de la comunicación con los interesados.
- **Desarrolladores de Software (2 personas):** El equipo de desarrollo se compone por cuatro programadores, divididos en:
 - **Desarrolladores Frontend:** Encargados de construir y optimizar la interfaz de usuario de la plataforma, garantizando que sea funcional y atractiva.

- **Desarrolladores Backend:** Responsables de desarrollar la lógica del servidor, integrar las bases de datos y asegurar la estabilidad y escalabilidad del sistema.
- **Analista de Sistemas (1 persona):** Se encarga de recopilar y analizar los requisitos del sistema, diseñando la estructura funcional del software y asegurando que se ajusta a las necesidades del usuario final. Además, colaborará con los desarrolladores para asegurar que el sistema cumpla con los estándares de calidad y funcionalidad requeridos.
- **Especialista en UX (1 persona):** Es responsable del diseño de la experiencia de usuario (UX), enfocándose en la creación de una interfaz intuitiva y accesible para todos los usuarios. Trabaja en la investigación de las necesidades de los usuarios, prototipado y pruebas de usabilidad, asegurando que la plataforma ofrezca una experiencia óptima.
- **Quality Assurance (QA) (1 personas):** Es responsable de realizar pruebas exhaustivas de la plataforma, tanto en términos funcionales como de rendimiento. Esto incluye pruebas de seguridad, usabilidad y rendimiento para garantizar que el producto final sea confiable, seguro y de alta calidad.

1.8.5. Planificación temporal, y asignación de recursos (Carta Gantt)

Para realizar este proyecto, emplearemos la metodología ágil tipo Scrum. Esta metodología promueve el desarrollo incremental del sistema mediante iteraciones denominadas releases las cuales contienen un conjunto de tareas. En este contexto, se planificó el desarrollo en cuatro lanzamientos, cada uno enfocado en la implementación de un módulo clave del sistema:

- Primera Release: Módulo de Solicitudes de Servicio
- Segunda Release: Módulo de Propuestas Profesionales
- Tercera Release: Módulo de Gestión de Servicios
- Cuarta Release: Módulo de Administración del Sistema

Los tiempos de desarrollo y los recursos asignados a cada módulo se ven reflejados en la siguiente tabla:

Tabla 1.8. Programación temporal del proyecto

Nombre de tarea	Duración	Costo	Nombres de los recursos
➤ PLATAFORMA WEB DE REPARACIONES DOMICILIARIAS	146 días	382,60 UF	
Gestión del proyecto	14 días	61,60 UF	JP;AS[50%]
➤ Primera Release: Módulo de Solicitudes de Servicio	36 días	102,84 UF	
Recolección y análisis de requerimientos	7 días	39,20 UF	AS;JP
Diseño de arquitectura y base de datos	8 días	14,72 UF	DV[50%];JP[20%]
Diseño de interfaz inicial	7 días	14,84 UF	UX[50%];AS[25%];JP[10%]
Desarrollo módulo de solicitudes	7 días	19,04 UF	JP[10%];DV
Pruebas funcionales y QA	4 días	4,48 UF	QA[50%];JP[10%]
Liberación + retroalimentación	3 días	10,56 UF	JP;QA[20%]
➤ Segunda Release: Módulo de Propuestas Profesionales	26 días	63,60 UF	
Recolección y análisis de requisitos	6 días	16,80 UF	JP[50%];AS[50%]
Diseño interfaz profesional	6 días	12,72 UF	UX[50%];AS[25%];JP[10%]
Desarrollo módulo de propuestas	7 días	19,04 UF	JP[10%];DV
Pruebas de seguridad y QA	4 días	4,48 UF	QA[50%];JP[10%]
Liberación + retroalimentación	3 días	10,56 UF	JP;QA[20%]
➤ Tercera Release: Módulo de Gestión de Servicios	26 días	63,20 UF	
Análisis y diseño de gestión de servicios	6 días	12,96 UF	AS[50%];JP[30%]
Desarrollo del flujo completo de servicio	10 días	27,20 UF	JP[10%];DV
Pruebas de usabilidad	7 días	13,44 UF	UX[50%];QA[25%];JP[10%]
Liberación + retroalimentación	3 días	9,60 UF	JP
➤ Cuarta Release: Módulo de Administración del Sistema	21 días	47,36 UF	
Análisis y diseño de reportes y paneles	6 días	12,96 UF	AS[50%];JP[30%]
Desarrollo módulo de administración	10 días	27,20 UF	JP[10%];DV
Pruebas funcionales e informes finales	5 días	7,20 UF	QA[50%];JP[20%]
Cierre del Proyecto	10 días	44,00 UF	JP;AS[50%]

Fuente: Elaboración propia.

Donde:

- AS: Analista de sistema
- DV: Desarrolladores de Software
- JP: Jefe de Proyecto
- UX: Especialista en UX
- QA: Quality Assurance

1.8.6. Administración del riesgo

En este punto se detalla los tipos de riesgos a los cuales se expuso el proyecto al ser desarrollado propiamente. Esto nos muestra la posibilidad que pase el riesgo, el impacto que tendrá en el proyecto, el costo que tendrá y qué prioridad tiene, el cual está ordenada de menor a mayor.

Tabla 1.9. Evaluación de Riesgos

Tipo	Número	Riesgo	Posibilidad	Impacto	Costo	Prioridad
Funcionalidad	1	El proyecto presenta defectos y errores en la implementación de las funcionalidades	7	8	1	12
Tecnológico	2	Fallas en la integración de tecnologías clave (APIs, bases de datos)	6	9	2	20
Personal	3	Falta de personal calificado durante la etapa de desarrollo, lo que retrasa el proyecto	8	7	3	24
Financiero	4	Sobrecostos inesperados que superan el presupuesto asignado al proyecto	5	9	4	24
Logística	5	Retrasos en la entrega de recursos o insumos críticos para el desarrollo del sistema	7	6	2	28
Seguridad	6	Vulnerabilidades detectadas durante la fase de pruebas, afectando la integridad de los datos de los usuarios	5	10	3	36
Cumplimiento	7	El proyecto no cumple con las normativas legales locales de protección de datos	4	9	4	28
Operativo	8	Interrupciones en el proceso de desarrollo debido a fallos en las herramientas de trabajo	6	7	2	36
Crecimiento	9	Problemas de escalabilidad del sistema a medida que crece el número de usuarios y transacciones	6	8	3	36
Reputación	10	Mala percepción de la plataforma por parte de los usuarios debido a errores o mala experiencia en el uso	5	7	2	42

Fuente: Elaboración propia

Teniendo en cuenta los riesgos se podrá crear un plan de mitigación y se asignará a uno o más responsables, los cuales se harán cargo de ejecutar cuando algún riesgo se haya presentado.

Tabla 1.10. Plan de mitigación

Número	Prioridad	Señal de Alerta	Responsable	Plan Preventivo	Plan Reactivo
1	12	Al implementar el proyecto en el entorno de producción, se encuentran problemas que afectan su funcionalidad	Desarrolladores de Software	Implementar pruebas exhaustivas en un entorno controlado antes de desplegar en producción	Aplicar procedimientos de reversión o parches rápidos para corregir los errores detectados post-despliegue
2	20	Errores recurrentes en la integración con sistemas externos (APIs)	Desarrolladores de Software	Realizar pruebas de integración durante el desarrollo y asegurar compatibilidad entre tecnologías	Realizar pruebas de integración durante el desarrollo y asegurar compatibilidad entre tecnologías
3	24	Sobrecarga en el equipo de desarrollo o tareas sin asignar por falta de personal	Jefe de Proyecto, Recursos Humanos	Establecer un plan de contratación anticipada y contar con recursos freelance en reserva	Reasignar tareas, priorizar entregas críticas y contratar apoyo temporal de emergencia
4	24	Desviaciones en el presupuesto asignado para el desarrollo del proyecto	Jefe de Proyecto, Financiero	Implementar un control estricto del presupuesto con revisiones periódicas y un fondo de contingencia	Ajustar el alcance del proyecto o reprogramar entregas para adaptarse al nuevo presupuesto disponible
5	28	Retrasos en la recepción de insumos o licencias críticas para el desarrollo	Jefe de Proyecto	Planificar las adquisiciones con antelación y mantener proveedores alternos identificados	Reasignar tareas que no dependan del insumo retrasado o negociar extensiones de plazo con stakeholders
6	36	Resultados de pruebas de seguridad muestran vulnerabilidades	Quality Assurance (QA), Desarrolladores de Software	Realizar auditorías de seguridad periódicas y aplicar buenas prácticas desde el diseño	Corregir vulnerabilidades críticas con parches de emergencia y restringir accesos hasta resolver
7	28	Falta de cumplimiento con normativas locales de protección de datos	Asesor Legal, Desarrolladores de Software	Trabajar con expertos legales desde las primeras fases del proyecto para asegurar cumplimiento normativo	Aplicar medidas correctivas recomendadas por auditorías legales y actualizar políticas de privacidad
8	36	Herramientas de trabajo inestables o caídas durante el desarrollo	Jefe de Proyecto, Desarrolladores de Software	Implementar redundancias o herramientas de respaldo para evitar interrupciones prolongadas	Cambiar temporalmente de herramienta o restaurar desde respaldos hasta resolver la falla
9	36	El sistema experimenta lentitud o problemas al aumentar la carga de usuarios	Desarrolladores de Software	Diseñar el sistema con escalabilidad desde el inicio y realizar pruebas de rendimiento	Habilitar servidores adicionales temporalmente o limitar nuevas solicitudes hasta estabilizar el sistema
10	42	Comentarios negativos recurrentes de usuarios sobre la experiencia de uso	Equipo de Marketing, Desarrolladores de Software	Realizar encuestas de satisfacción y monitorear feedback de usuarios	Aplicar mejoras rápidas de usabilidad y publicar actualizaciones informativas que aclaren errores

Fuente: Elaboración propia

1.8.7. Estimación de costos.

La estimación de costos por cada recurso se basa en la distribución de trabajo y las tarifas definidas previamente en este capítulo. A partir de estos datos, se calcula el costo total del proyecto considerando el esfuerzo total (en horas) de cada recurso y su tarifa horaria. En la siguiente tabla se presenta la estimación detallada:

Tabla 1.11. Estimación de costos por recurso

Recurso	Sueldo Mensual (UF)	Trabajo	Costo Total (UF)	Costo/Hr	Max. Disp.
Jefe de Proyecto	62	541,6 Hrs	209,2	0,4	100%
Analista de Sistemas	52	160 Hrs	52,0	0,3	100%
Especialista en UX	45	80 Hrs	22,5	0,3	100%
Desarrolladores	45	304 Hrs	85,2	0,3	200%
QA	30	75,6 Hrs	21,0	0,2	100%

Fuente: Elaboración propia

El porcentaje de disponibilidad (“Max. Disp.”) indica la proporción de tiempo asignado a cada recurso en el proyecto. Por ejemplo, los desarrolladores tienen un 200% de disponibilidad porque el rol está asignado a dos personas. Como resultado, el costo total estimado de los recursos es de 389,9 UF. Al sumar un 5% por riesgos y un 20% de utilidades, el costo total del proyecto asciende a 497,3 UF, con un tiempo estimado de ejecución de 29 semanas.

CAPÍTULO 2. ASPECTOS RELEVANTES DEL ANÁLISIS PARA LA ALTERNATIVA SELECCIONADA

2. ASPECTOS RELEVANTES DEL ANÁLISIS.

En este capítulo, se realiza un análisis en profundidad de la alternativa seleccionada para el desarrollo del proyecto. Se incluye una descripción detallada de la solución propuesta, acompañada de un diagrama de actividades general. Además, se especifican los requisitos del sistema, distinguiendo entre requisitos funcionales y no funcionales. Se aborda también la estructura funcional del sistema, la matriz de trazabilidad, el modelo conceptual, y el modelo de casos de uso, proporcionando así una comprensión integral de la propuesta técnica.

2.1. Descripción de la solución propuesta

La solución que se propone consiste en el desarrollo de una herramienta digital que permitirá a los propietarios de viviendas cargar solicitudes de servicio a través de un formulario detallado en la plataforma. Este formulario incluirá campos para especificar el tipo de problema, detalles de la ubicación, fotos o videos del área afectada, y cualquier comentario adicional relevante para los profesionales. Una vez completadas, estas solicitudes serán enviadas automáticamente a una red de profesionales o técnicos especializados, quienes podrán revisar, evaluar y cotizar el costo de la reparación o mantenimiento requerido.

El sistema contará con un mecanismo de notificaciones en tiempo real para que los propietarios reciban actualizaciones sobre el estado de sus solicitudes, así como propuestas enviadas por los profesionales. Cada propuesta incluirá detalles sobre el tiempo estimado de finalización, costo total, materiales requeridos (si corresponde) y una breve descripción de la metodología que se aplicará para resolver el problema.

El objetivo principal de la plataforma es facilitar la interacción entre los propietarios de viviendas y los profesionales de manera eficiente y transparente. Esto se logrará mediante funcionalidades adicionales como:

- **Sistema de comparación de propuestas:** Los propietarios podrán comparar las propuestas recibidas basándose en criterios como precio, tiempo de realización y calificaciones previas del profesional.
- **Sistema de calificación y reseñas:** Una vez completado el trabajo, los propietarios podrán dejar una reseña y calificar al profesional, lo cual ayudará a futuros usuarios a tomar decisiones informadas por medio de esta retroalimentación.
- **Historial de solicitudes:** Los usuarios tendrán acceso a un historial detallado de todas las solicitudes realizadas, con el fin de mantener un registro de las reparaciones y servicios previos.

- **Seguridad y protección de datos:** La plataforma garantizará la privacidad de la información proporcionada por los usuarios, de acuerdo con las normativas locales e internacionales de protección de datos.
- **Soporte multicanal:** El sistema contará con un servicio de atención al cliente que ayudará a los usuarios a resolver cualquier duda o problema que surja durante el uso de la plataforma.

Para los profesionales, la plataforma no solo representará un medio para ofrecer sus servicios, sino también una herramienta para gestionar su trabajo de manera más eficiente, dado que podrán recibir notificaciones sobre nuevas solicitudes, gestionar su calendario de trabajos, y recibir pagos seguros a través de la plataforma, con la opción de recibir el pago directamente a sus cuentas bancarias.

La solución también integrará un panel administrativo para la gestión y monitoreo del funcionamiento del sistema. Este panel permitirá a los administradores de la plataforma supervisar el flujo de solicitudes y ofertas, verificar la calidad de los servicios y garantizar que los usuarios cumplan con las normas de la comunidad.

El objetivo de esta solución es ofrecer una experiencia completa tanto para los propietarios que buscan servicios de reparación como para los profesionales que buscan expandir sus servicios y captar más clientes. Al implementar esta solución, se espera optimizar y agilizar la forma en que se gestionan y resuelven los problemas en los hogares, fomentando la transparencia de los servicios.

2.2. Diagrama de actividades general UML (flujo administrativo)

En este punto se presenta un análisis detallado del flujo administrativo que se lleva a cabo dentro del sistema propuesto. Para comprender el funcionamiento del sistema, es necesario entender la importancia del diagrama de actividades para saber cómo interactúan los diferentes actores involucrados en el sistema (clientes, sistema y profesionales) y cómo se desarrolla cada etapa del proceso. El propósito de este diagrama es visualizar de manera clara y estructurada el recorrido de la solicitud desde su creación hasta su finalización.

El flujo administrativo comienza cuando un cliente registra una solicitud de servicio en la plataforma, proporcionando detalles específicos sobre el problema que necesita resolver. Este proceso desencadena una serie de validaciones automáticas y acciones por parte del sistema y de los profesionales registrados. Este diagrama permite identificar posibles cuellos de botella y facilita la comprensión de cómo se gestionan las propuestas, la selección del profesional y la interacción post-servicio para asegurar la satisfacción del cliente y la mejora continua del sistema.

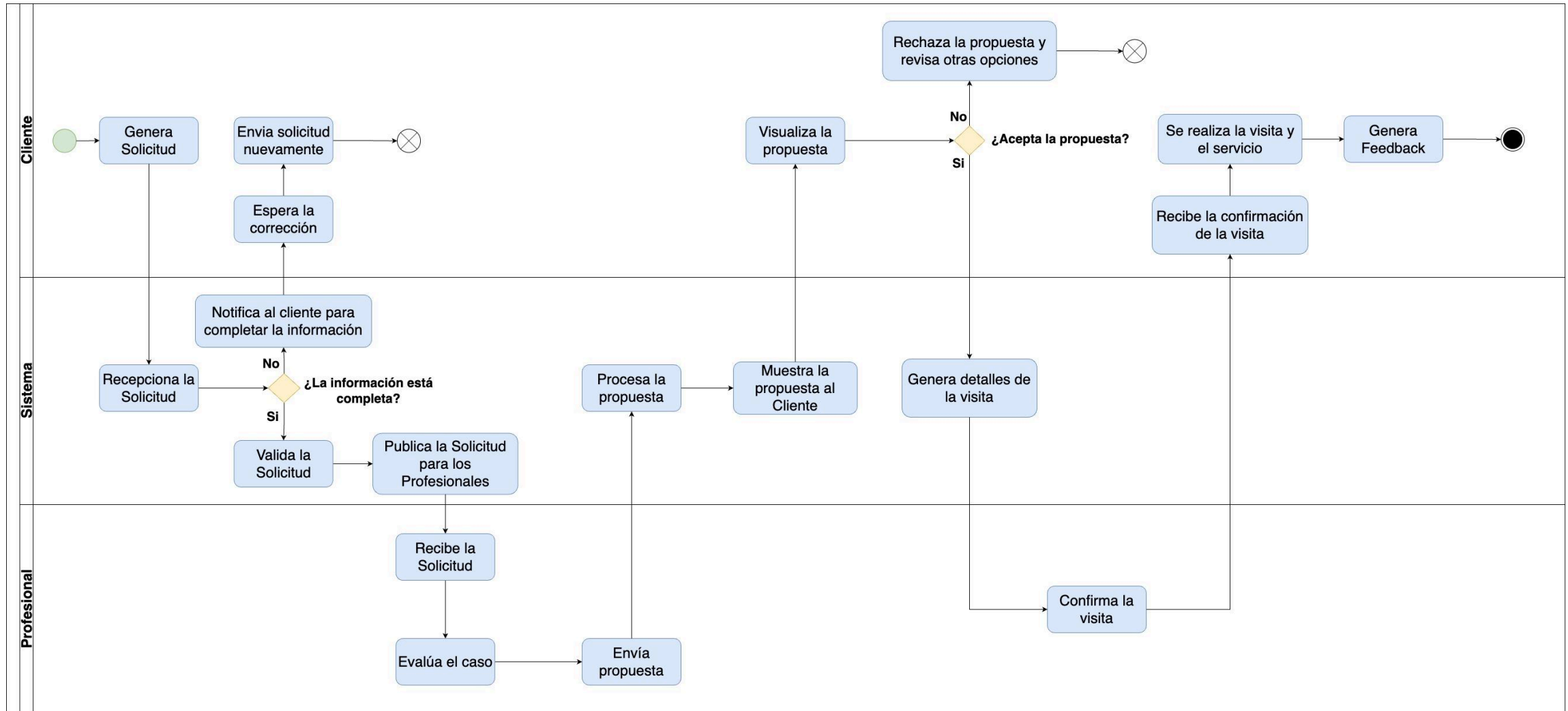


Figura 2.1. Diagrama de Actividad UML del Flujo de Solicitud de Servicio

Fuente: Elaboración propia

El diagrama de actividades presentado en la Figura 2.1 demuestra de manera detallada el proceso administrativo desde la creación de una solicitud de servicio por parte de un cliente hasta su finalización con la generación de retroalimentación. Este diagrama permite visualizar las interacciones entre los tres principales actores del sistema, los cuales son: el cliente, el sistema y el profesional.

El flujo de trabajo comienza cuando un cliente accede a la plataforma y genera una solicitud describiendo el problema que necesita resolver en su hogar. Esta solicitud es recibida por el sistema, el cual inicia un proceso de validación para asegurar que la información proporcionada esté completa. Si la solicitud está completa, el sistema continúa con su validación para proceder a su publicación. Sin embargo, si la información está incompleta, el sistema notifica al cliente y solicita la corrección de los datos. Una vez que el cliente completa la información y reenvía la solicitud, el sistema la recibe nuevamente y repite el proceso de validación.

Cuando la solicitud es validada y aprobada, el sistema la publica y notifica a los profesionales registrados en la plataforma. Los profesionales que estén interesados en la solicitud evalúan el caso y envían una propuesta al cliente que incluye detalles como el costo estimado, el tiempo de ejecución y los aspectos técnicos del servicio ofrecido.

Una vez que el cliente visualiza las propuestas recibidas, tiene la opción de aceptarlas o rechazarlas. Si el cliente acepta una de las propuestas, el sistema genera automáticamente los detalles de la visita y notifica al profesional seleccionado. Si el cliente decide rechazar las propuestas, el flujo se cierra en ese punto, con la posibilidad de que el cliente revise otras opciones o espere nuevas propuestas de otros profesionales.

El siguiente paso en el proceso involucra al profesional que, al confirmar la visita, él se prepara para realizar el trabajo en la ubicación indicada por el cliente. Tras la realización del servicio, el sistema notifica al cliente para que el cliente pueda evaluar el resultado y proporcionar una retroalimentación. Esta etapa final es crucial, ya que permite al cliente calificar al profesional y la calidad del trabajo realizado, contribuyendo a un sistema de evaluación que beneficia tanto a futuros usuarios como a los profesionales, fomentando la transparencia de la plataforma.

2.1 Requerimientos del sistema

A continuación, se detallan los requerimientos del sistema, obtenidos a partir de los requerimientos de usuario y el diagrama de actividades presentado anteriormente. Estos requerimientos están divididos en dos categorías, los funcionales y los no funcionales, cada uno es

esencial para garantizar el correcto funcionamiento de la plataforma y el cumplimiento de los objetivos del proyecto.

2.3.1 Requerimientos funcionales

En este punto se presentan los requerimientos funcionales, los cuales describen las funcionalidades esenciales que el sistema debe cumplir para garantizar su correcto funcionamiento y satisfacer las necesidades de los usuarios. Estos requerimientos se enfocan en detallar las acciones específicas que debe realizar la plataforma, abarcando desde la creación de solicitudes hasta la gestión de propuestas, visitas y evaluaciones.

Tabla 2.1. Requerimientos Funcional

RF	Descripción
RF-01	El sistema debe permitir al Cliente registrar solicitudes de servicio, especificando detalles como descripción, imágenes y categoría.
RF-02	El sistema debe validar la información de cada solicitud y publicarla para que los Profesionales puedan visualizarla.
RF-03	El sistema debe permitir al Cliente cancelar solicitudes antes de que se seleccione alguna propuesta.
RF-04	El sistema debe permitir al Cliente visualizar, comparar, aceptar o rechazar las propuestas enviadas por los Profesionales.
RF-05	El sistema debe permitir a los Usuarios consultar el historial de solicitudes y servicios realizados, incluyendo fechas y estados.
RF-06	El sistema debe permitir a Clientes y Profesionales actualizar sus perfiles, modificando información personal y de contacto.
RF-07	El sistema debe permitir al Cliente realizar evaluaciones y calificaciones de los Profesionales una vez concluido el servicio.
RF-08	El sistema debe permitir al Profesional marcar el servicio como finalizado al concluir su ejecución.
RF-09	El sistema debe permitir al Profesional responder a las solicitudes publicadas, enviando información sobre costos, tiempos y recursos.
RF-10	El sistema debe permitir al Profesional gestionar la creación y el desarrollo de los servicios, coordinando fechas y actualizando sus estados hasta su finalización.
RF-11	El sistema debe permitir al Administrador gestionar notificaciones, definiendo mensajes y reglas de envío automático a los usuarios.
RF-12	El sistema debe permitir al Administrador crear, modificar, bloquear y reactivar cuentas de Clientes y Profesionales.
RF-13	El sistema debe permitir al Administrador intervenir y resolver disputas reportadas entre Clientes y Profesionales.
RF-14	El sistema debe permitir al Administrador revisar y, en caso necesario, modificar o cancelar solicitudes que infrinjan normas o políticas.
RF-15	El sistema debe permitir al Administrador obtener informes sobre la actividad y estadísticas generales, como solicitudes activas, servicios concluidos y niveles de satisfacción.

Fuente: Elaboración propia

2.3.2 Requerimientos no funcionales

Los requerimientos no funcionales establecidos en este punto definen las características de calidad y los criterios técnicos que debe cumplir la plataforma para asegurar un desempeño eficiente, seguro y compatible con el entorno tecnológico. Estos requerimientos complementan a los funcionales, garantizando que el sistema sea confiable, escalable y fácil de mantener.

Tabla 2.2. Requerimientos No Funcionales

RNF	Descripción	Medio de Verificación
RNF-01	El sistema debe garantizar que la comunicación entre cliente y servidor esté cifrada mediante HTTPS, y que las contraseñas sean almacenadas de forma segura utilizando hashing y salting.	Certificado SSL activo y revisión de los métodos de hashing en auditorías semestrales.
RNF-02	El sistema debe ser capaz de manejar un aumento del 50% en usuarios concurrentes sin afectar el rendimiento.	Pruebas de carga, esto medirá el tiempo de respuesta esperando que sea < 2 segundos con 500 usuarios concurrentes.
RNF-03	El sistema debe permitir la implementación de actualizaciones y parches sin interrumpir el servicio durante más de 30 minutos.	Tiempo promedio de despliegue medido en pruebas de mantenimiento.
RNF-04	El sistema debe proporcionar una interfaz intuitiva y accesible desde dispositivos móviles y navegadores modernos, permitiendo a un usuario nuevo completar una solicitud sin necesidad de capacitación previa.	Evaluación mediante la prueba de usabilidad estándar System Usability Scale. Se espera un puntaje mínimo de 80 sobre 100, correspondiente a una usabilidad de nivel “Excelente”.
RNF-05	El sistema debe ser compatible con las versiones más recientes de navegadores (Chrome, Firefox, Safari, Edge, etc.) y sistemas operativos (Windows, Android, MacOS e iOS).	Pruebas de compatibilidad realizadas en todos los navegadores y sistemas operativos mencionados.
RNF-06	El sistema debe garantizar la consistencia de los datos en todas las transacciones, evitando duplicaciones o pérdidas de información.	Porcentaje de transacciones exitosas (meta: 99.9%).
RNF-07	El sistema debe cumplir con las normativas locales, como la Ley de Protección de Datos Personales (Ley 19.628 Sobre Protección de la Vida Privada), e incluir mecanismos para que los usuarios gestionen sus datos personales.	Auditorías de cumplimiento realizadas de forma semestral.
RNF-08	El sistema debe registrar al menos el 95% de las transacciones en tiempo real y mantener logs durante un período mínimo de 6 meses.	Monitoreo de logs con herramientas de análisis (Elastic Stack, Graylog, Datadog, etc.).

Fuente: Elaboración propia

2.2 Estructura funcional

En esta sección se describe la estructura funcional del sistema, organizada mediante un diagrama de caso de uso general que identifica las interacciones entre los actores principales y las funcionalidades del sistema. A continuación, se detallan los actores involucrados y su rol dentro del sistema:

- **Cliente:** Usuario que registra solicitudes de servicios, selecciona las propuestas enviadas por profesionales, coordina visitas y evalúa los servicios realizados. Además, recibe notificaciones del sistema sobre el estado de sus solicitudes. Se representa de color azul.
- **Profesional:** Usuario que evalúa las solicitudes publicadas, envía propuestas detalladas y confirma las visitas asignadas. También recibe notificaciones del sistema relacionadas con las solicitudes en las que participó.
- **Administrador:** Usuario con privilegios avanzados que supervisa el sistema, genera reportes, administra notificaciones y administra usuarios.

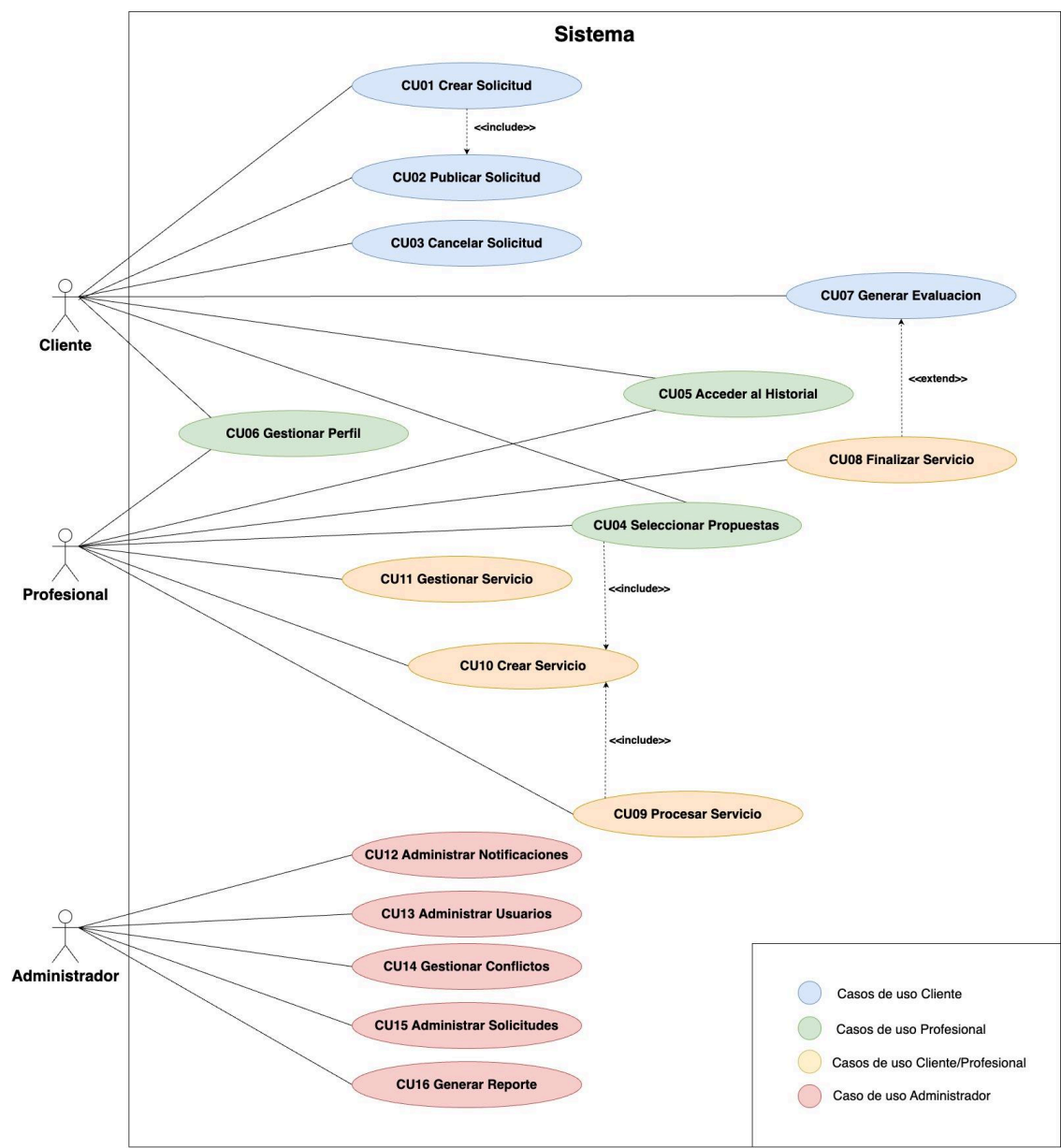


Figura 2.2 Caso de uso general
Fuente: Elaboración propia

Donde

El sistema cuenta con casos de uso principales que reflejan las funcionalidades básicas requeridas para cumplir con los objetivos del proyecto. En el Diagrama de Caso de Uso General se visualizan las siguientes interacciones clave:

- **CU01 Crear Solicitud:** Permite registrar una nueva solicitud de servicio, detallando la descripción del problema, adjuntando imágenes y especificando la categoría.
- **CU02 Publicar Solicitud:** Valida la información de la solicitud y se publica para que los profesionales puedan enviar propuestas.
- **CU03 Cancelar Solicitud:** Ofrece la posibilidad de cancelar una solicitud antes de que se seleccione alguna propuesta en caso de que ya no sea necesario contar con el servicio.
- **CU04 Seleccionar Propuestas:** Proporciona la opción de visualizar y comparar las propuestas enviadas por los profesionales, con el fin de elegir la más conveniente de acuerdo con criterios como costo y tiempo.
- **CU05 Acceder al Historial:** Permite consultar un historial completo de solicitudes y servicios realizados, tanto para clientes como para profesionales, incluyendo fechas, estados y detalles relevantes.
- **CU06 Gestionar Perfil:** Facilita la modificación de información personal para clientes y profesionales, como datos de contacto o preferencias de la cuenta.
- **CU07 Generar Evaluación:** Permite la evaluación y calificación del profesional una vez finalizado el servicio, aportando retroalimentación que favorece la transparencia del sistema. Esta evaluación se visualiza luego en el historial.
- **CU08 Finalizar Servicio:** Permite que el profesional indique la finalización del trabajo, habilitando así la opción de evaluación por parte del cliente.
- **CU09 Procesar Servicio** Ofrece a los profesionales la funcionalidad de responder a las solicitudes publicadas, especificando costos, plazos y recursos necesarios. Forma parte del flujo de generación y presentación de propuestas.
- **CU10 Crear Servicio:** Inicia el proceso de prestación del servicio al aceptarse una propuesta. Asocia la solicitud aprobada con la ejecución formal de esta y asocia la solicitud aprobada con su ejecución formal.
- **CU11 Gestionar Servicio:** Facilita la supervisión del servicio en ejecución, con acciones como coordinación de fechas, actualizando estados y gestionando posibles modificaciones a los requerimientos. Este proceso está vinculado directamente con la creación previa del servicio.

- **CU12 Administrar Notificaciones:** Permite al administrador configurar la forma en que se envían notificaciones automáticas a los distintos tipos de usuarios, así como establecer campañas o mensajes específicos.
- **CU13 Administrar Usuarios:** Brinda la posibilidad de gestionar las cuentas del sistema, incluyendo la creación, modificación, bloqueo o eliminación de usuarios, conforme las políticas establecidas.
- **CU14 Gestionar Conflictos:** Facilita la resolución de disputas entre clientes y profesionales. El administrador evalúa la situación y aplica medidas de mediación o sanciones, si corresponde.
- **CU15 Administrar Solicitudes:** Permite al administrador revisar, modificar o cancelar solicitudes que puedan infringir normas o requerir supervisión administrativa.
- **CU16 Generar Reporte:** Ofrece la elaboración de reportes sobre la actividad en el sistema, incluyendo estadísticas de solicitudes, servicios finalizados y niveles de satisfacción de los usuarios.

2.3 Matriz de trazabilidad

Para garantizar que todos los requerimientos funcionales del sistema sean cubiertos de manera integral, se utiliza la Matriz de Trazabilidad. Esta herramienta relaciona los requerimientos funcionales con los casos de uso, asegurando que cada requerimiento esté asociado a al menos un caso de uso y que cada caso de uso cumpla con uno o más requerimientos. Lo anterior permite verificar la cobertura total de los objetivos del sistema, asegurando su coherencia y robustez. La matriz no solo es esencial en la fase de diseño y desarrollo, sino que también resulta fundamental durante las etapas de pruebas y validación, garantizando que todas las funcionalidades requeridas sean implementadas y evaluadas correctamente.

Tabla 2.3. Matriz de trazabilidad del sistema

Caso de uso/ Requerimiento Funcional	CU01	CU02	CU03	CU04	CU05	CU06	CU07	CU08	CU09	CU10	CU11	CU12	CU13	CU14	CU15	CU16
RF-01	x															
RF-02		x														
RF-03			x													
RF-04				x												
RF-05					x											
RF-06						x										
RF-07							x									
RF-08								x								
RF-09									x							
RF-10										x	x					
RF-11												x				
RF-12													x			
RF-13														x		
RF-14															x	
RF-15																x

Fuente: Elaboración propia

La Matriz de Trazabilidad es una herramienta esencial para verificar la cobertura completa de los requerimientos funcionales a través de los casos de uso definidos en el sistema. Esta matriz garantiza que cada funcionalidad requerida esté correctamente respaldada por una interacción específica dentro del diseño del sistema. Al establecer una relación directa entre los requerimientos funcionales y los casos de uso, permite identificar de manera sistemática cualquier posible vacío en lo que cubrirá entre Caso de Uso y Requerimiento Funcional, asegurando que el sistema cumpla con los objetivos del proyecto de manera integral y coherente.

Además, esta matriz proporciona al equipo de desarrollo una guía clara durante las fases de diseño e implementación, asegurando que se consideren todos los requerimientos del cliente y que no se omita ningún aspecto crítico.

2.4 Modelo Conceptual

A continuación, se presenta el modelo conceptual del sistema de gestión de reparaciones y mantenimiento en el hogar. Este modelo considera las entidades clave, sus relaciones y la cardinalidad entre ellas, lo cual facilita una estructura sólida para el almacenamiento y gestión de los datos en el sistema. Se destaca el rol del usuario "Administrador", quien es responsable de gestionar el sistema, garantizando su funcionamiento adecuado y la asignación eficiente de profesionales a las solicitudes de servicio.

La entidad principal del modelo es "Solicitud", que actúa como el núcleo del sistema al concentrar la mayoría de las relaciones y permitir la vinculación entre clientes, profesionales y servicios. Adicionalmente, el modelo incluye las entidades "Cliente", "Profesional", "Servicio", "Evaluación" y "Calificación", cada una con relaciones específicas que reflejan un sistema integral que cubre todos los aspectos necesarios para la administración de las reparaciones y mantenciones en el hogar.

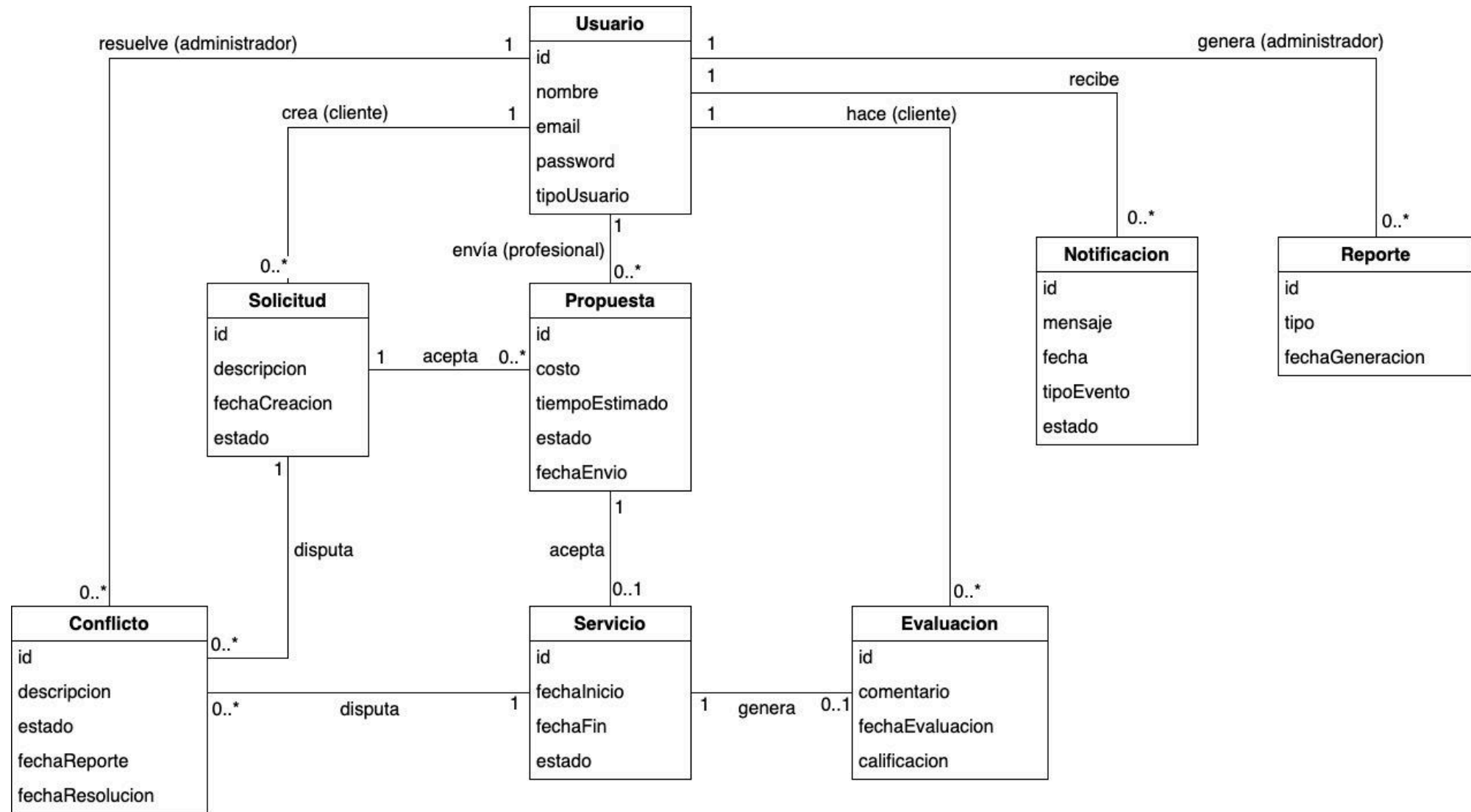


Figura 2.3. Modelo conceptual

Fuente: Elaboración propia

2.5 Modelo de Casos de Uso

En este apartado se muestran todos los casos de uso narrativos con sus diagramas de secuencia y los contratos respectivos.

2.5.1 Caso de uso: CU01- Crear Solicitud

Tabla 2.4. Caso de uso narrativo: CU01- Crear Solicitud

Caso de uso:	Crear Solicitud
Código:	CU01
Actor:	Cliente
Descripción:	El cliente proporciona los datos necesarios para registrar una solicitud de servicio en el sistema, incluyendo detalles del problema, imágenes y la categoría.
Referencias cruzadas:	CU02 Publicar Solicitud
Precondición:	El cliente debe haber iniciado sesión en el sistema y debe tener acceso al formulario de creación de solicitudes.
Postcondición:	Se registró una nueva solicitud en el sistema con los datos proporcionados, y quedó lista para su validación y publicación.
Flujo Principal	
Actor	Sistema
1. El cliente accede a la plataforma	
	2. El sistema despliega menú de inicio
3. El cliente selecciona la opción "Crear Solicitud".	
	4. El sistema despliega el formulario de creación de solicitudes.
5. El cliente completa el formulario con la descripción del problema, imágenes/videos, categoría y se envía.	
	6. El sistema valida que todos los campos requeridos están completos, correctos
	7. El sistema crea la solicitud con estado "pendiente de publicación".
	8. El sistema despliega por pantalla la confirmación de la creación de la solicitud

Fuente: Elaboración propia

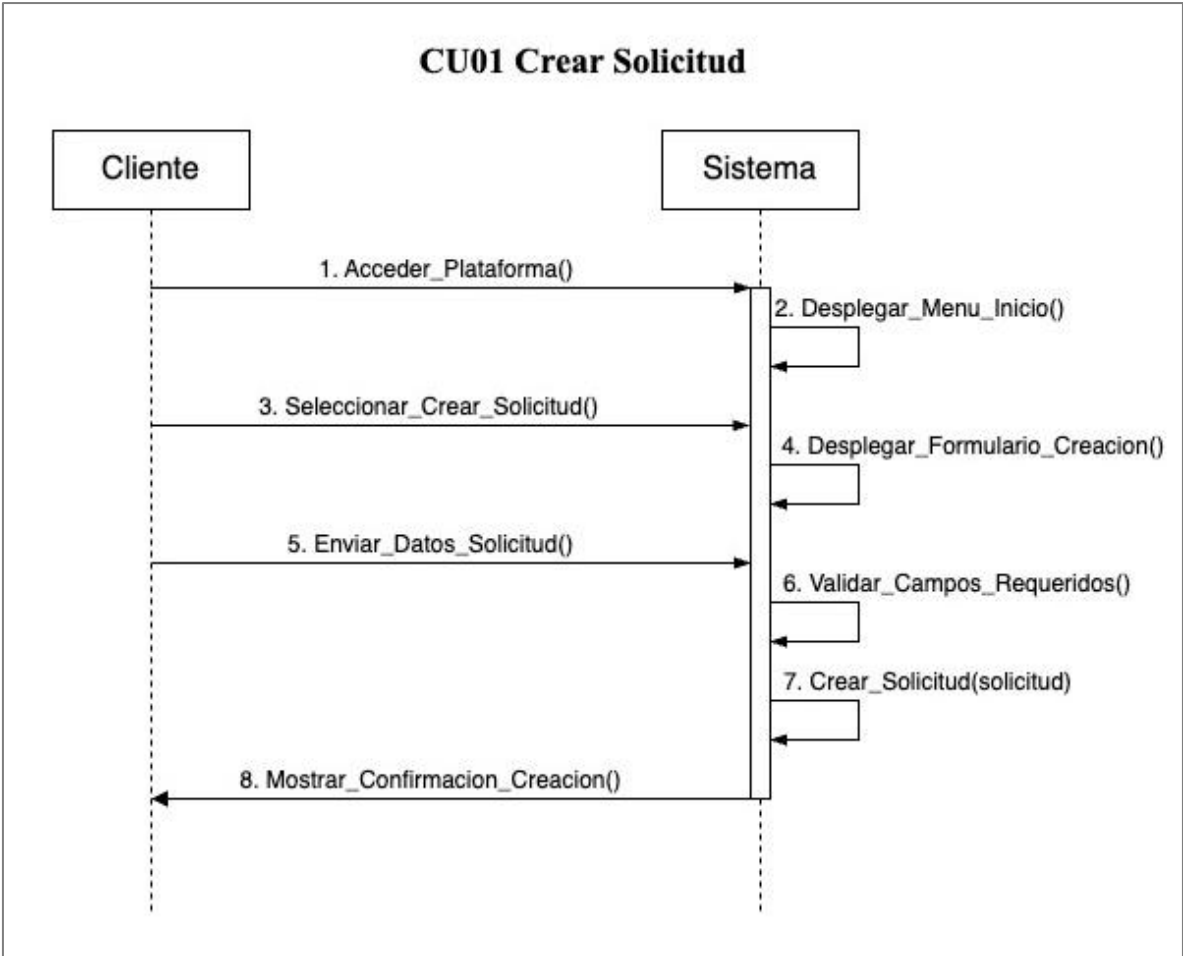


Figura 2.4. Diagrama de secuencia: CU01- Crear Solicitud
Fuente: Elaboración propia

Tabla 2.5. Contrato: CU01- Crear Solicitud

Nombre	Crear_Solicitud(solicitud)
Ref. Cruzada	No tiene.
Precondición	El cliente debe haber iniciado sesión y tener acceso al formulario de creación de solicitudes, además de haber completado y validado todos los campos requeridos antes de llamar a este método.
Postcondición	Se registró una nueva Solicitud con estado “pendiente de publicación”, asociada al cliente que la generó, quedando disponible para su posterior publicación y mostrándose un mensaje de confirmación al cliente sobre su creación exitosa.

Fuente: Elaboración propia

2.5.2 Caso de uso: CU02 - Publicar Solicitud

Tabla 2.6. Caso de uso narrativo: CU02 - Publicar Solicitud

Caso de uso:	Publicar Solicitud
Código:	CU02
Actor:	Usuario: Profesionales, Clientes
Descripción:	El sistema se encarga de publicar las solicitudes que han sido validadas, dejándolas a disposición de los profesionales registrados para que puedan enviar sus propuestas. Este proceso forma parte del flujo de creación de solicitudes.
Referencias cruzadas:	CU01 Crear Solicitud
Precondición:	La solicitud debe estar validada previamente por el sistema y lista para su publicación.
Postcondición:	La solicitud quedó publicada en la plataforma y disponible para los profesionales registrados.
Flujo Principal	
Actor	Sistema
	1. El sistema recibe la solicitud validada desde el flujo del caso de uso "Crear Solicitud".
	2. El sistema valida por medio de entidades externas los campos relevantes pertenecientes a los datos del usuario.
	3. El sistema asigna el estado de "Publicada" a la solicitud y queda visible para los profesionales en la plataforma, junto con todos los detalles necesarios para que puedan evaluarla y enviar propuestas.
	4. El sistema notifica a los profesionales registrados sobre la nueva solicitud publicada.
Flujo Alternativo: Datos Inconsistentes	
	2a. El sistema detecta una inconsistencia con los datos ingresados.
	2b. El sistema notifica al cliente y la solicitud cambia a estado "Rechazado".

Fuente: Elaboración propia

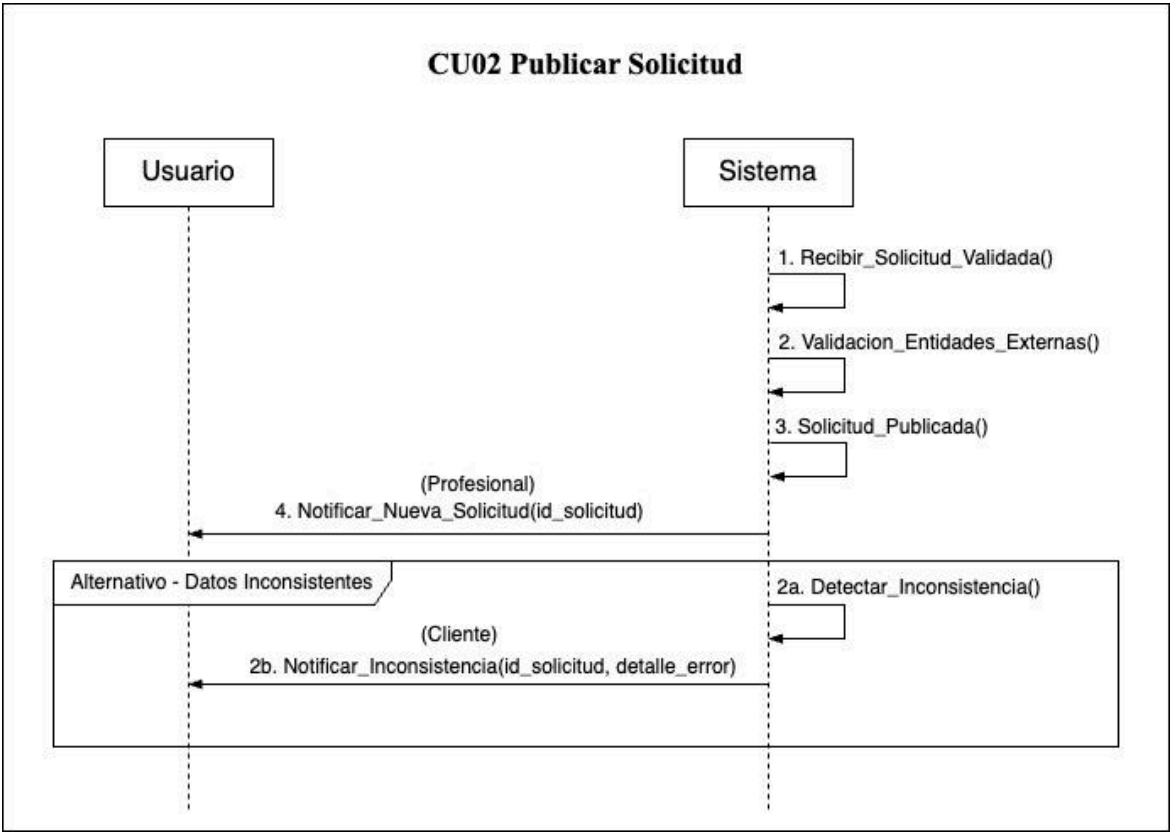


Figura 2.5. Diagrama de secuencia 02 - Publicar Solicitud
Fuente: Elaboración propia

Tabla 2.7. Contrato: CU02 - Publicar Solicitud

Nombre	Solicitud_Publicada(id_solicitud)
Ref. Cruzada	No tiene.
Precondición	La solicitud ha sido validada previamente, se encuentra lista para su publicación y está registrada en la base de datos con un estado “pendiente de publicación”.
Postcondición	La solicitud cambió su estado a “publicada” y quedó disponible para los profesionales en la plataforma. El sistema notificó a los profesionales registrados sobre la nueva solicitud publicada y el Cliente la vio marcada como “publicada”.

Fuente: Elaboración propia

2.5.3 Caso de uso: CU03 - Cancelar Solicitud

Tabla 2.8. Caso de uso narrativo: CU03 - Cancelar Solicitud

Caso de uso:	Cancelar Solicitud
Código:	CU03
Actor:	Cliente
Descripción:	El sistema permite al Cliente cancelar una de sus solicitudes previamente creadas y que aún no haya sido cerrada o finalizada. Una vez cancelada, la solicitud ya no estará disponible para el envío de propuestas ni para su posterior publicación.
Referencias cruzadas:	No tiene.
Precondición:	Existe una solicitud en estado “pendiente de publicación” o “publicada” asociada al Cliente; no debe estar “finalizada” ni “cancelada” previamente.
Postcondición:	La solicitud pasa a estado “cancelada” y, si estaba “publicada”, deja de mostrarse a los Profesionales. En caso de estar “pendiente”, simplemente se descarta y no podrá publicarse más.
Flujo Principal	
Actor	Sistema
1. El Cliente accede al listado de solicitudes que se encuentren creadas.	2. El sistema muestra la lista de solicitudes que el Cliente puede cancelar y recibe la selección de la solicitud específica a cancelar.
3. El Cliente confirma la cancelación de la solicitud.	4. El sistema verifica el estado de la solicitud (que no esté finalizada o ya cancelada).
	5. El sistema cambia el estado de la solicitud a “cancelada”.
	6. El sistema informa al Cliente que la solicitud fue cancelada con éxito.
Flujo Alternativo: Estado Solicitud Incompatible	
	4a. El sistema detecta que el estado de la solicitud es “finalizada” o “cancelada”.
	4b. El sistema despliega por pantalla un mensaje sobre que la solicitud no es posible de cancelar por el estado de esta.

Fuente: Elaboración propia

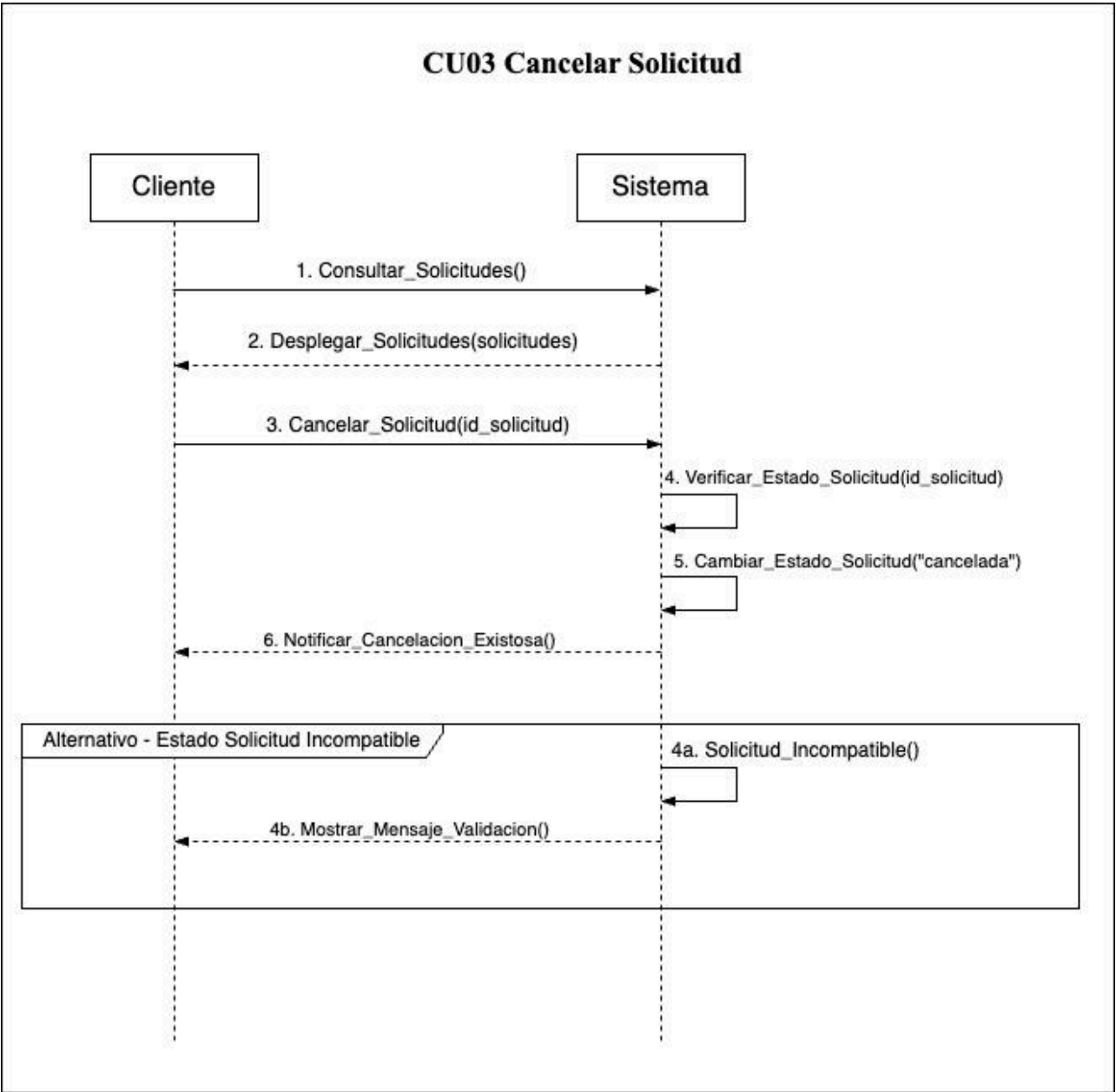


Figura 2.6. Diagrama de secuencia: CU03 - Cancelar Solicitud
Fuente: Elaboración propia

Tabla 2.9. Contrato: CU03 - Cancelar Solicitud

Nombre	Cambiar_Estado_Solicitud("cancelada")
Ref. Cruzada	No tiene.
Precondición	Existe una solicitud en estado “pendiente” o “publicada” asociada al Cliente, y no se encuentra “finalizada” ni “cancelada” previamente. Además, el Cliente ha confirmado la acción de cancelación.
Postcondición	La solicitud cambió su estado a “cancelada” y dejó de estar disponible a los profesionales en la plataforma. El cliente recibió una confirmación de que la cancelación se realizó con éxito.

Fuente: Elaboración propia

2.5.4 Caso de uso: CU04 - Seleccionar Propuestas

Tabla 2.10. Caso de uso narrativo: CU04 - Seleccionar Propuestas

Caso de uso:	Seleccionar Propuestas
Código:	CU04
Actor:	Usuario: Cliente, Profesional
Descripción:	El sistema permite al Cliente visualizar las propuestas que han sido enviadas por los Profesionales para una solicitud en estado “publicada” y seleccionar aquella que considere más adecuada. Una vez seleccionada, dicha propuesta cambia a un estado de “aceptación”, y el sistema notifica al Profesional para dar continuidad al proceso de servicio.
Referencias cruzadas:	CU09 Procesar Servicio, CU10 Crear Servicio
Precondición:	Solicitud publicada con propuestas enviadas por profesionales.
Postcondición:	Se registró la propuesta seleccionada por el cliente y se notificó al profesional correspondiente.
Flujo Principal	
Actor	Sistema
1. El Cliente accede a la lista de propuestas para su solicitud publicada.	2. El sistema muestra la información de cada propuesta (costo, tiempo estimado, detalles del profesional, etc.) asociada a la solicitud seleccionada.
3. El Cliente elige la propuesta que desea aceptar.	4. El sistema verifica que la propuesta siga vigente y que la solicitud continúe “publicada”.
	5. El sistema cambia el estado de la propuesta a “aceptada”, y las demás propuestas quedan en estado “rechazado”.
	6. El sistema notifica al Profesional correspondiente que su propuesta ha sido seleccionada.
	7. El sistema informa al Cliente que la selección se completó satisfactoriamente.
Flujo Alternativo: Estado Propuesta Incompatible	
	4a. El sistema detecta que el estado de la solicitud es “finalizada” o “cancelada” y que la propuesta esté vigente.
	4b. El sistema despliega por pantalla un mensaje sobre que la propuesta ya no es válida y el motivo.

Fuente: Elaboración propia

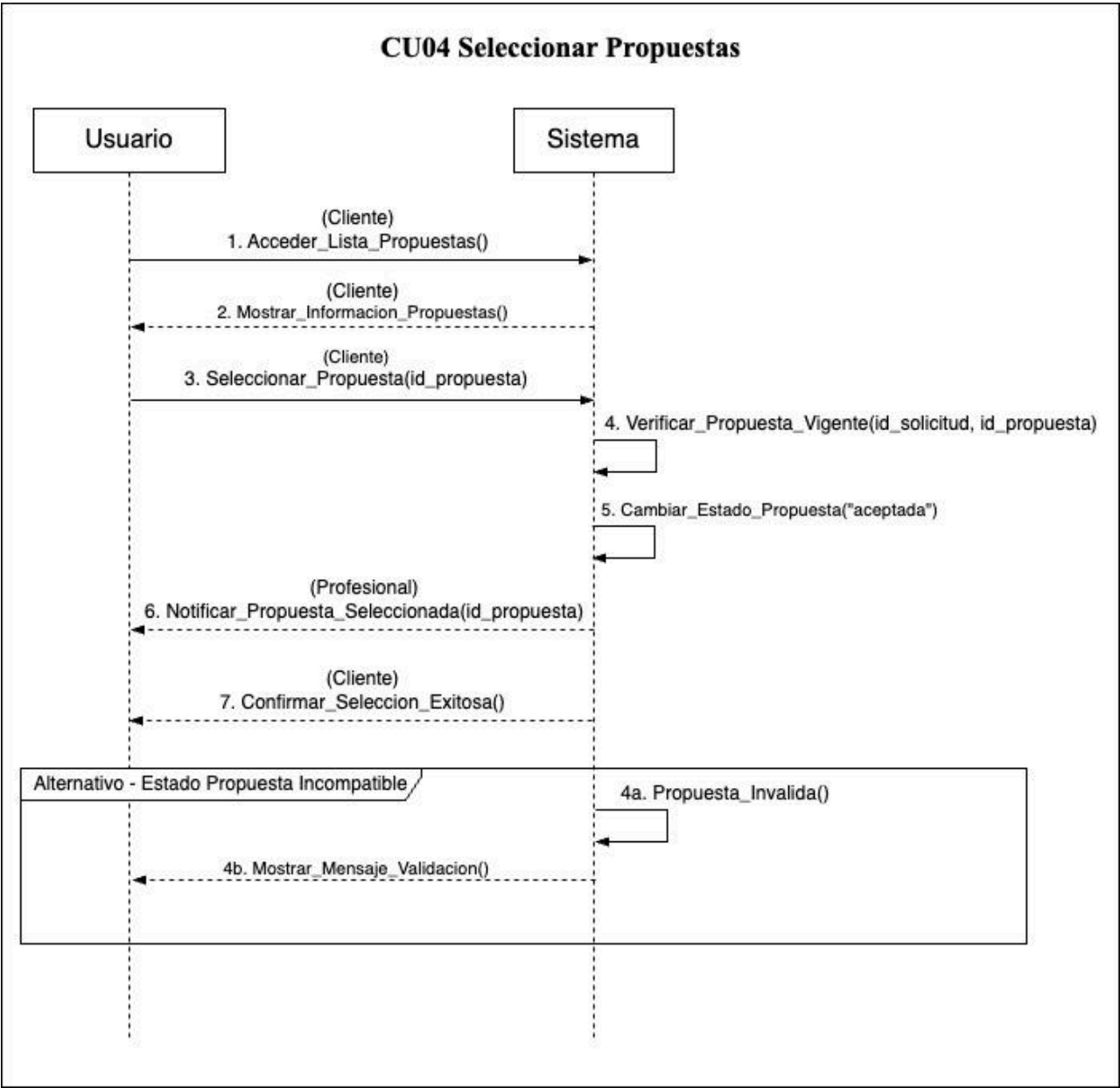


Figura 2.7. Diagrama de secuencia 04 - Seleccionar Propuestas
Fuente: Elaboración propia

Tabla 2.11. Contrato 04 - Seleccionar Propuestas

Nombre	Cambiar_Estado_Propuesta("aceptada")
Ref. Cruzada	CU09 Procesar Servicio, CU10 Crear Servicio
Precondición	Existe al menos una propuesta vigente para la solicitud publicada, y el Cliente ha elegido la propuesta que desea aceptar. Además, la verificación de vigencia confirma que la propuesta sigue válida.
Postcondición	La propuesta seleccionada cambió su estado a “aceptada”, a su vez las demás propuestas de la misma solicitud pasaron a “rechazadas” y el Profesional fue notificado de que su propuesta ha sido elegida.

Fuente: Elaboración propia

2.5.5 Caso de uso: CU05 - Acceder Historial

Tabla 2.12. Caso de uso narrativo 05 Acceder Historial

Caso de uso:	Acceder Historial
Código:	CU05
Actor:	Usuario: Cliente, Profesional
Descripción:	Permite al Usuario consultar un historial completo de solicitudes o servicios realizados incluyendo fechas, estados y otros detalles. El Usuario selecciona el tipo de historial que desea ver (por ejemplo, "Solicitudes" o "Servicios") y el sistema devuelve la lista correspondiente, además de la opción de descargar como reporte.
Referencias cruzadas:	CU07 Generar Evaluación
Precondición:	El Usuario ha iniciado sesión y existe al menos un registro (solicitud o servicio) que pueda visualizarse en el historial.
Postcondición:	El Usuario visualiza en pantalla la lista (solicitudes o servicios) asociada al tipo de historial que seleccionó, con posibilidad de ver detalles de cada ítem (fecha, estado, profesional, etc.) y la funcionalidad de descargar como el historial tipo reporte.
Flujo Principal	
Actor	Sistema
1. El Usuario accede al módulo de historial.	
	2. Muestra la lista con las diferentes opciones para generar el historial.
3.- El Usuario selecciona la opción.	
	4. Muestra el detalle del historial seleccionado en pantalla y la opción “Descargar”
5.- Selecciona “Descargar Historial”	
	6.- Mostrar descarga exitosa

Fuente: Elaboración propia

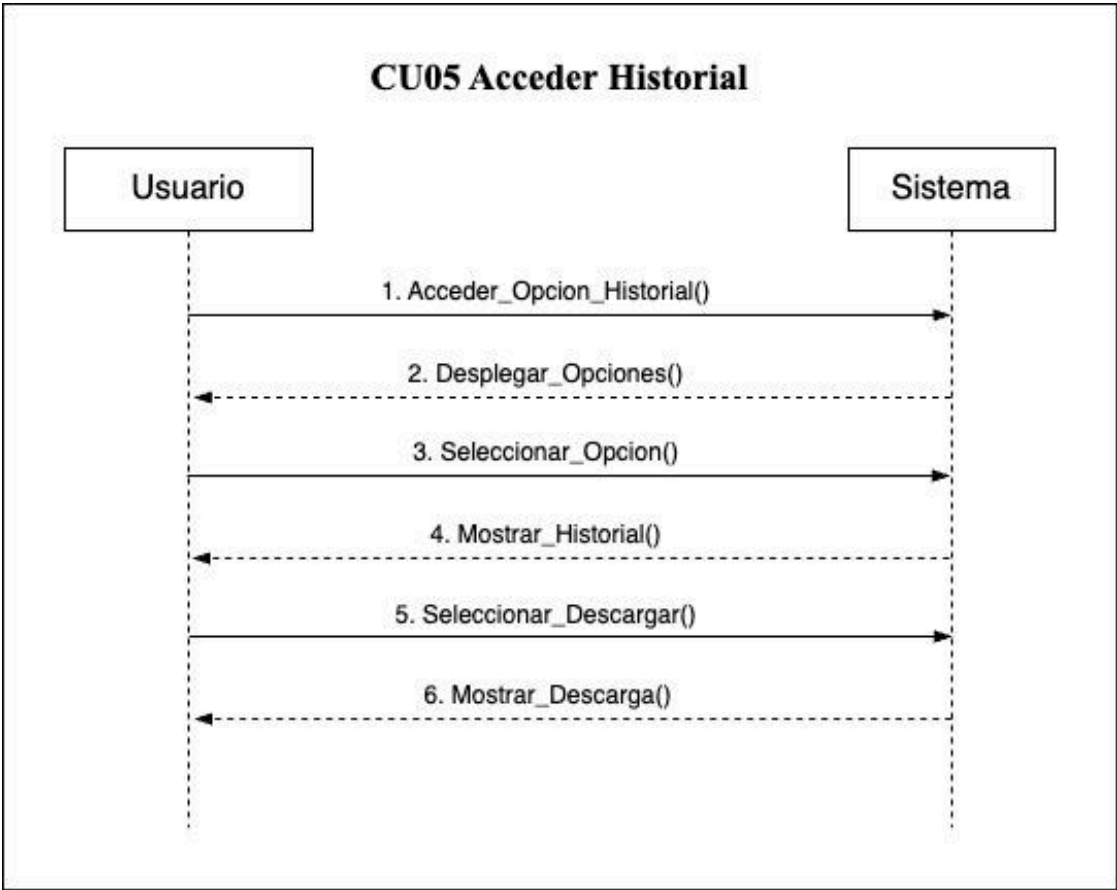


Figura 2.8. Diagrama de secuencia: CU05 - Acceder Historial
Fuente: Elaboración propia

Tabla 2.13. Contrato: CU05 - Acceder Historial

Nombre	Seleccionar_Descargar()
Ref. Cruzada	CU07 Generar Evaluación
Precondición	El Usuario (Cliente o Profesional) ya visualiza el historial (solicitudes o servicios) y se le ha mostrado la opción “Descargar”. Además, se encuentra autenticado en la plataforma.
Postcondición	Se inició la descarga del historial seleccionado y el sistema mostró un mensaje confirmando que la finalizó con éxito.

Fuente: Elaboración propia

2.5.6 Caso de uso: CU06 - Gestionar Perfil

Tabla 2.14. Caso de uso narrativo: CU06 - Gestionar Perfil

Caso de uso:	Gestionar Perfil	
Código:	CU06	
Actor:	Usuario: Cliente, Profesional	
Descripción:	Permite al Usuario modificar su información personal.	
Referencias cruzadas:	No tiene.	
Precondición:	El Usuario debe haber iniciado sesión en la plataforma.	
Postcondición:	La información del perfil del Usuario queda actualizada en el sistema.	
Flujo Principal		
Actor	Sistema	
1. El Usuario accede a la sección “Gestionar Perfil” en la plataforma.		
	2. El sistema carga y muestra los datos actuales del perfil para su edición.	
3. El Usuario modifica uno o varios campos.		
	4. El sistema valida los datos ingresados para ser actualizar el perfil.	
	5. El sistema informa al Usuario que los cambios fueron guardados exitosamente y los datos del perfil quedan reflejados inmediatamente.	

Fuente: Elaboración propia

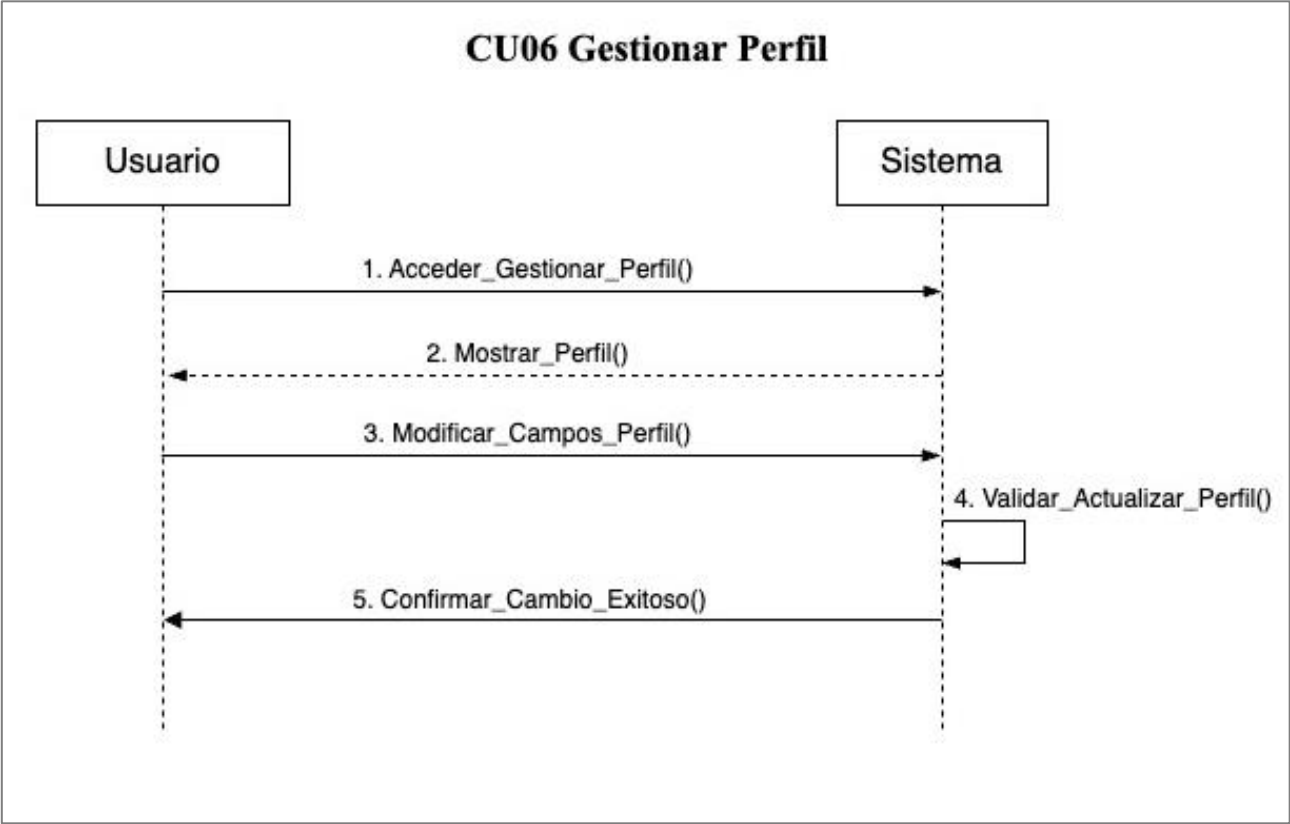


Figura 2.9. Diagrama de secuencia: CU06 - Gestionar Perfil
Fuente: Elaboración propia

Tabla 2.15. Contrato: CU06 - Gestionar Perfil

Nombre	Validar_Actualizar_Perfil()
Ref. Cruzada	No tiene.
Precondición	El Usuario (Cliente o Profesional) debe haber iniciado sesión, accedido a la sección “Gestionar Perfil” y modificado uno o varios campos de su información personal en la interfaz antes de llamar a este método.
Postcondición	La información del perfil del Usuario quedó actualizada en el sistema; los cambios se reflejaron de inmediato, y el sistema confirmó que se guardaron correctamente.

Fuente: Elaboración propia

2.5.7 Caso de uso: CU07 - Generar Evaluación

Tabla 2.16. Caso de uso narrativo: CU07 - Generar Evaluación

Caso de uso:	Generar Evaluación
Código:	CU07
Actor:	Cliente
Descripción:	Permite al Cliente calificar y dejar retroalimentación (comentarios, puntuación, etc.) sobre un servicio que ya ha sido finalizado, de este modo el profesional recibe una evaluación y el sistema pueda reflejar esa información a futuros usuarios.
Referencias cruzadas:	CU05 Acceder al Historial
Precondición:	El Cliente debe tener al menos un servicio finalizado asociado a su cuenta, además, el Cliente debe haber iniciado sesión para poder evaluar.
Postcondición:	Se registra la evaluación en el sistema, vinculada al servicio y al profesional correspondiente, quedando lista para futuras consultas o referencias.
Flujo Principal	
Actor	Sistema
1. El Cliente ingresa a la sección “Servicios”.	
	2. El sistema muestra la lista de servicios que el Cliente tiene asociados.
3. El Cliente filtra por sus servicios finalizados.	
	4. El sistema presenta la lista filtrada de servicios que ya han sido marcados como finalizados.
5. El Cliente selecciona uno de los servicios finalizados para evaluarlo.	
	6. El sistema despliega un formulario de evaluación (puntuación, comentarios, etc.).
7. El Cliente completa y envía la información de la evaluación.	
	8. El sistema valida los datos, registra la evaluación y confirma al Cliente que la evaluación se guardó exitosamente.

Fuente: Elaboración propia

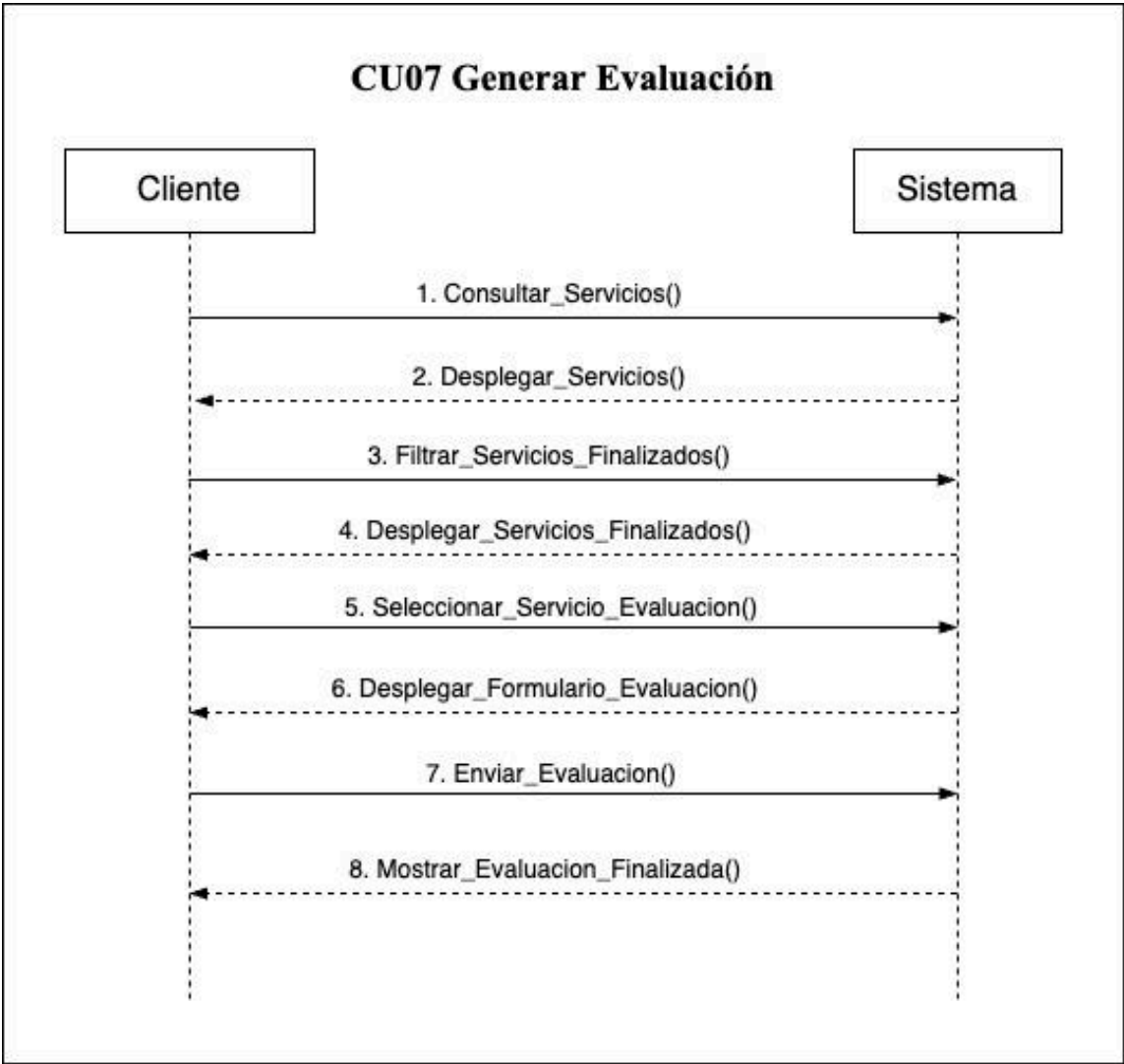


Figura 2.10. Diagrama de secuencia: CU07 - Generar Evaluación
Fuente: Elaboración propia

Tabla 2.17. Contrato: CU07 - Generar Evaluación

Nombre	Enviar_Evaluacion()
Ref. Cruzada	CU05 Acceder al Historial
Precondición	El Cliente ha filtrado y seleccionado un servicio finalizado, y el sistema ha mostrado el formulario de evaluación, con campos para calificación y comentarios.
Postcondición	Se registró la nueva evaluación asociada al servicio y al profesional que lo realizó, quedando lista para ser consultada en el historial o la información del profesional. El sistema confirmó al Cliente el éxito de la evaluación.

Fuente: Elaboración propia

2.5.8 Caso de uso: CU08 - Finalizar Servicio

Tabla 2.18. Caso de uso narrativo: CU08 - Finalizar Servicio

Caso de uso:	Finalizar Servicio
Código:	CU08
Actor:	Profesional
Descripción:	El sistema permite al Profesional indicar que un servicio en curso ha sido completado. Una vez marcado como “Finalizado”, el Cliente podrá posteriormente evaluarlo y así cerrar el ciclo del servicio.
Referencias cruzadas:	CU07 Generar Evaluación
Precondición:	El Profesional debe haber iniciado sesión y existir al menos un servicio en estado “Activo” que pueda ser finalizado.
Postcondición:	El servicio cambia a estado “Finalizado”, el sistema notifica al Profesional la confirmación y habilita al Cliente para que pueda evaluar el trabajo realizado.
Flujo Principal	
Actor	Sistema
1. El Profesional ingresa a “Mis Servicios” en la plataforma.	
	2. El sistema despliega la lista de servicios que el Profesional tiene en estado “Activo”.
3. El Profesional selecciona uno de los servicios en curso para marcarlo como finalizado.	
	4. El sistema solicita confirmación de la finalización.
5. El Profesional confirma la finalización.	
	6. El sistema cambia el estado del servicio a “Finalizado” y actualiza la base de datos.
	7. El sistema informa al Profesional que el servicio se ha marcado como finalizado y, si corresponde, habilita al Cliente para la evaluación.

Fuente: Elaboración propia

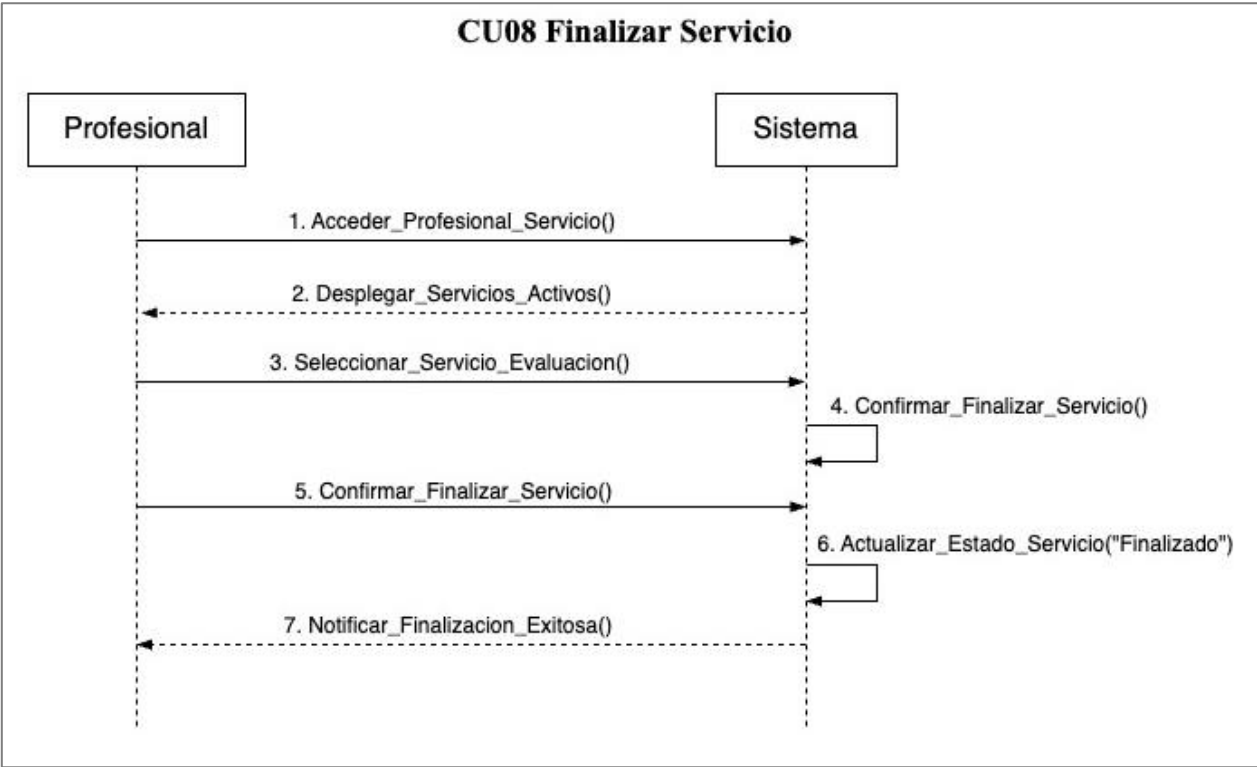


Figura 2.11. Diagrama de secuencia: CU08 - Finalizar Servicio
Fuente: Elaboración propia

Tabla 2.19. Contrato: CU08 - Finalizar Servicio

Nombre	Cambiar_Estado_Servicio("finalizado")
Ref. Cruzada	CU07 Generar Evaluación
Precondición	El Profesional debe haber seleccionado un servicio “Activo” y confirmado la finalización, garantizando que el servicio esté listo para finalizar.
Postcondición	Se actualizó el estado del servicio a “finalizado”, y el sistema notificó al Profesional que el cambio fue exitoso. Además, si corresponde, se habilitó la opción para que el Cliente realice la evaluación del servicio.

Fuente: Elaboración propia

2.5.9 Caso de uso: CU09 – Procesar Servicio

Tabla 2.20. Caso de uso narrativo: CU09 – Procesar Servicio

Caso de uso:	Procesar Servicio
Código:	CU09
Actor:	Profesional
Descripción:	Permite al Profesional responder a una solicitud publicada enviando una propuesta que incluye costo estimado, tiempo de ejecución y un mensaje descriptivo. La propuesta queda asociada a la solicitud y visible para el Cliente, quien podrá aceptarla o rechazarla.
Referencias cruzadas:	CU04 Seleccionar Propuestas, CU10 Crear Servicio
Precondición:	El Profesional tiene que haber iniciado sesión y existe al menos una solicitud en estado “Publicada” disponible para recibir propuestas.
Postcondición:	La propuesta queda registrada en el sistema, vinculada a la solicitud y al profesional, con estado “Enviada”, lista para que el Cliente la evalúe.
Flujo Principal	
Actor	Sistema
1. El Profesional accede a la sección “Solicitudes Publicadas”.	
	2. El sistema muestra la lista de solicitudes disponibles.
3. El Profesional selecciona una solicitud para revisarla en detalle.	
	4. El sistema despliega la información completa de la solicitud y habilita la opción “Enviar Propuesta”.
5. El Profesional completa el formulario con costo estimado, plazo y mensaje y hace clic en “Enviar Propuesta”.	
	6. El sistema valida los datos ingresados y registra la propuesta con estado “Enviada”.
	7. El sistema confirma al Profesional que la propuesta fue enviada correctamente y queda visible para el Cliente.

Fuente: Elaboración propia

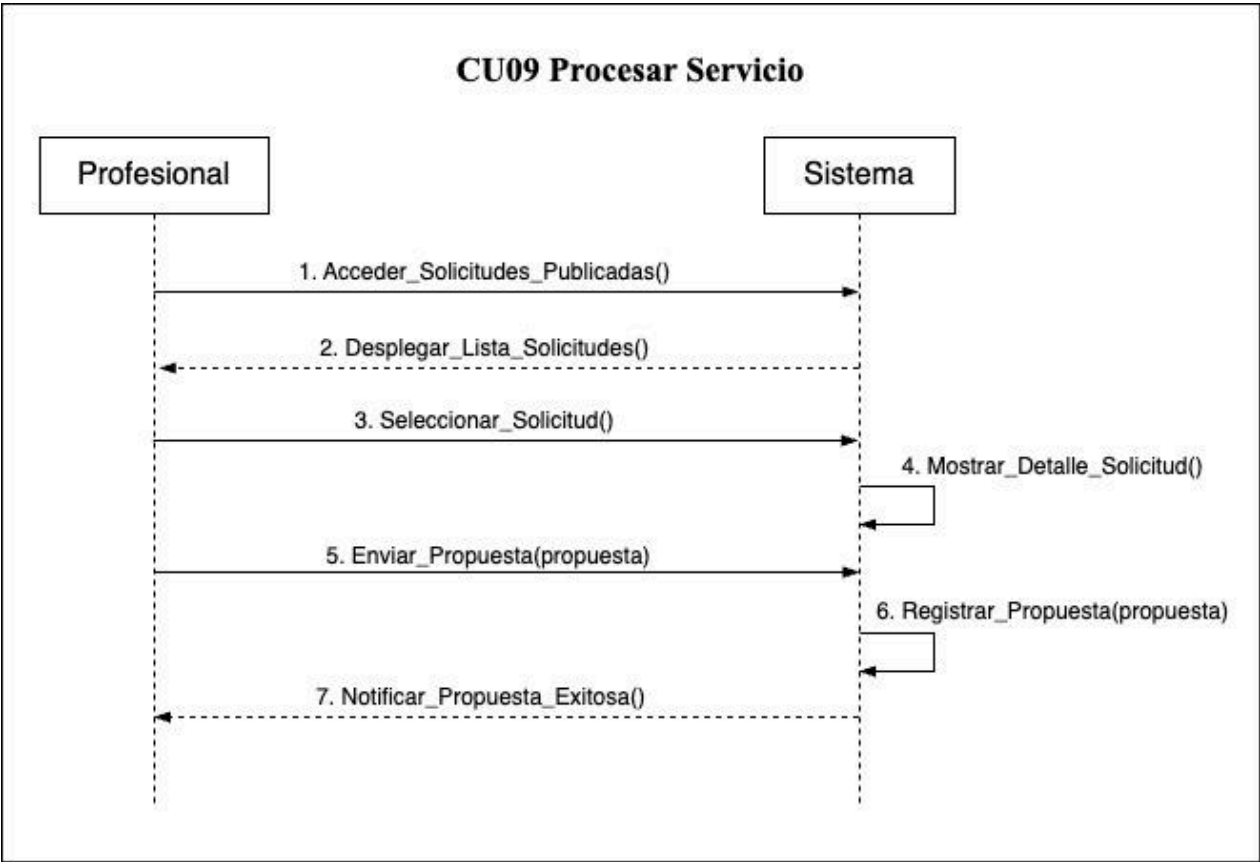


Figura 2.12. Diagrama de secuencia: CU09 – Procesar Servicio
Fuente: Elaboración propia

Tabla 2.21. Contrato: CU09 – Procesar Servicio

Nombre	Registrar_Propuesta(propuesta)
Ref. Cruzada	CU04 Seleccionar Propuestas, CU10 Crear Servicio
Precondición	El Profesional ha completado los datos de la propuesta y ha hecho clic en “Enviar Propuesta,” contando con una solicitud “Publicada” para asociar su oferta.
Postcondición	Se creó un nuevo registro de propuesta en la base de datos, asociado a la solicitud y al Profesional, con estado “Enviada.” El sistema notificó al Profesional que la propuesta fue enviada correctamente y quedó visible al Cliente.

Fuente: Elaboración propia

2.5.10 Caso de uso: CU10 – Crear Servicio

Tabla 2.22. Caso de uso narrativo: CU10 – Crear Servicio

Caso de uso:	Crear Servicio
Código:	CU10
Actor:	Usuario: Cliente, Profesional
Descripción:	Una vez que una propuesta ha sido aceptada. El sistema genera automáticamente un servicio asociado a la solicitud correspondiente y al profesional que envió la propuesta. El nuevo servicio se registra con estado “Activo” para permitir su seguimiento.
Referencias cruzadas:	CU04 Seleccionar Propuesta, CU11 Gestionar Servicio.
Precondición:	Debe existir una propuesta aceptada por el Cliente, vinculada a una solicitud “Publicada”.
Postcondición:	El servicio queda registrado en la base de datos con estado “Activo”, asociado al Profesional y al Cliente, y ambos reciben notificación de que el servicio ha sido creado.
Flujo Principal	
Actor	Sistema
	1. Detecta la propuesta marcada como “Aceptada”.
	2. Verifica que la solicitud y la propuesta estén en estados correctos para generar un servicio.
	3. Crea el registro de Servicio, lo asocia al Profesional y al Cliente, y lo inicializa con estado “Activo”.
	4. Notifica al Profesional que se ha generado un nuevo servicio asociado a su propuesta aceptada.
	5. Notifica al Cliente que el servicio se ha creado y está en curso.

Fuente: Elaboración propia

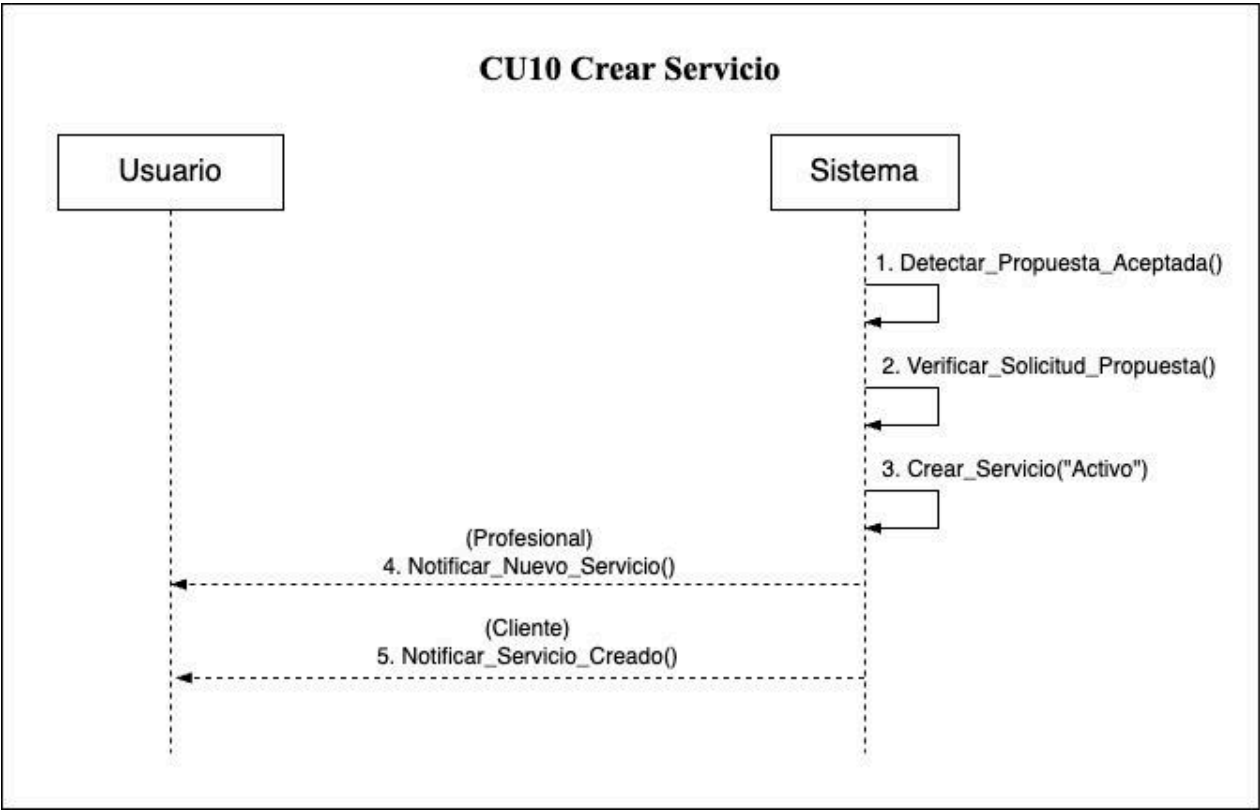


Figura 2.13. Diagrama de secuencia: CU10 – Crear Servicio
Fuente: Elaboración propia

Tabla 2.23. Contrato: CU10 – Crear Servicio

Nombre	Crear_Servicio("Activo")
Ref. Cruzada	CU04 Seleccionar Propuestas, CU11 Gestionar Servicio
Precondición	Se ha detectado una propuesta aceptada, verificado que la solicitud y la propuesta se encuentran en estados aptos para generar un servicio, y el sistema está listo para crear el servicio.
Postcondición	Se registró un nuevo Servicio con estado “Activo”, asociado al Profesional y al Cliente involucrados. El sistema notificó al Profesional y al Cliente que el servicio fue creado con éxito.

Fuente: Elaboración propia

2.5.11 Caso de uso: CU11 – Gestionar Servicio

Tabla 2.24. Caso de uso narrativo: CU11 – Gestionar Servicio

Caso de uso:	Gestionar Servicio
Código:	CU11
Actor:	Profesional
Descripción:	Permite al Profesional administrar un servicio que se encuentra en estado “Activo”. Incluye registrar avances, reprogramar fechas, subir evidencia (fotos, comentarios) y cambiar el estado del servicio.
Referencias cruzadas:	CU10 Crear Servicio
Precondición:	El Profesional ha iniciado sesión y tiene al menos un servicio en estado “Activo” asociado a su cuenta.
Postcondición:	El servicio queda actualizado con la información con el avance. Si el Profesional marcó el servicio como “Finalizado”, queda listo para la evaluación del Cliente.
Flujo Principal	
Actor	Sistema
1. El Profesional ingresa a la sección “Mis Servicios” en la plataforma.	2. El sistema muestra la lista de servicios en estado “Activo” asociados al Profesional.
3. El Profesional selecciona un servicio para gestionarlo.	4. El sistema despliega la ficha detallada del servicio, con opciones para actualizar avances, subir evidencias o reprogramar la fecha/hora.
5. El Profesional registra un avance (o reprograma la fecha) y guarda los cambios.	6. El sistema valida la información ingresada y actualiza el servicio en la base de datos con el nuevo estado o datos de avance.
	7. El sistema confirma al Profesional que los cambios fueron registrados correctamente y actualiza la vista con la información más reciente.
Flujo Alternativo: Datos Inconsistentes	
	6a. Muestra un mensaje de error indicando el problema.
	6b. Permite al Profesional corregir la información y volver a enviarla.

Fuente: Elaboración propia

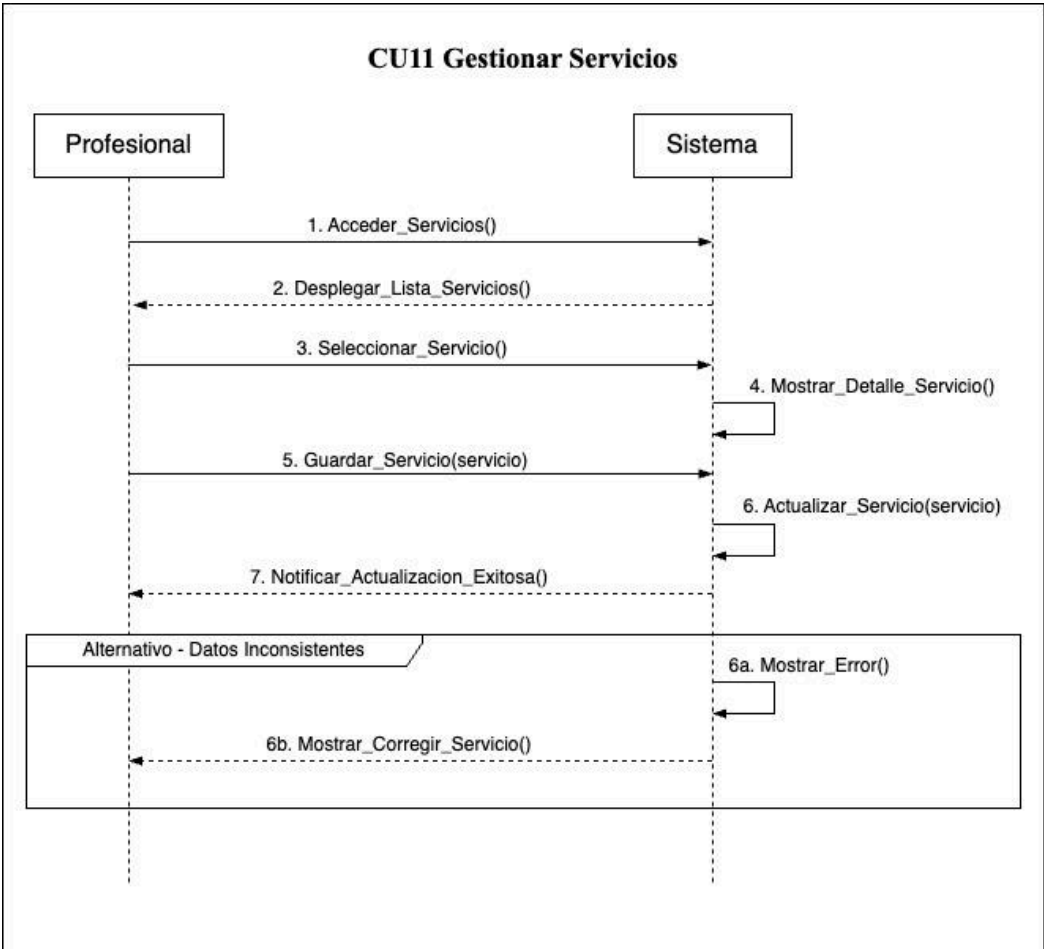


Figura 2.14. Diagrama de secuencia: CU11 – Gestionar Servicio
Fuente: Elaboración propia

Tabla 2.25. Contrato: CU11 – Gestionar Servicio

Nombre	Actualizar_Servicio(servicio)
Ref. Cruzada	CU10 Crear Servicio
Precondición	El Profesional está autenticado y ha seleccionado un servicio en estado “Activo”, ha visualizado su información detallada y ha guardado los cambios en la interfaz.
Postcondición	El servicio quedó actualizado en la base de datos con los datos de avance, reprogramación o estado que el Profesional modificó. El sistema confirmó al Profesional que la operación se realizó con éxito y refleja la información más reciente.

Fuente: Elaboración propia

2.5.12 Caso de uso: CU12 – Administrar Notificaciones

Tabla 2.26. Caso de uso narrativo: CU12 – Administrar Notificaciones

Caso de uso:	Administrar Notificaciones
Código:	CU12
Actor:	Administrador
Descripción:	Permite al Administrador crear, editar o eliminar mensajes de notificación globales (campañas o avisos) que el sistema enviará automáticamente a Clientes y/o Profesionales cuando se produzcan eventos clave. También puede activar o desactivar campañas existentes.
Referencias cruzadas:	No tiene.
Precondición:	El Administrador ha iniciado sesión y tiene privilegios para acceder al módulo de notificaciones.
Postcondición:	La configuración de la notificación (mensaje, tipo de destinatario, activado/desactivado) queda almacenada y lista para que el sistema la envíe automáticamente cuando ocurra el evento correspondiente.
Flujo Principal	
Actor	Sistema
1. El Administrador accede al módulo “Notificaciones” en la plataforma.	
	2. El sistema muestra la lista de campañas o mensajes existentes, con opciones para crear, editar, eliminar o activar/desactivar cada notificación.
3. El Administrador selecciona la opción deseada.	
	4. El sistema despliega el formulario correspondiente.
5. El Administrador completa la información y guarda los cambios.	
	6. El sistema valida los datos y guarda la notificación; luego confirma al Administrador que la operación se realizó exitosamente.
	7. El sistema informa al Administrador que la operación se realizó exitosamente.

Fuente: Elaboración propia

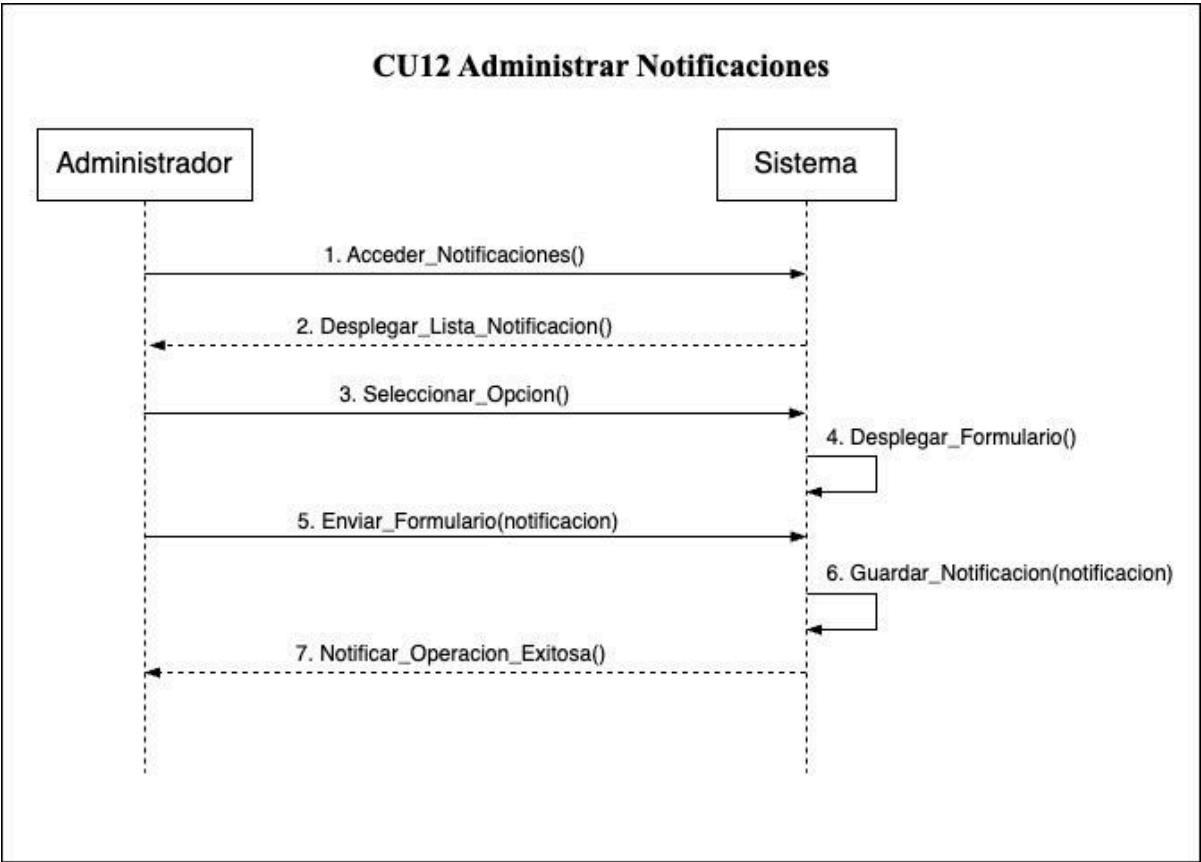


Figura 2.15. Diagrama de secuencia: CU12 – Administrar Notificaciones
Fuente: Elaboración propia

Tabla 2.27. Contrato: CU12 – Administrar Notificaciones

Nombre	Guardar_Notificacion(notificacion)
Ref. Cruzada	No tiene.
Precondición	El Administrador ha seleccionado la operación y completado el formulario con los datos de la notificación. Posee privilegios de administración.
Postcondición	La notificación quedó registrada/actualizada en la base de datos con el mensaje, tipo de destinatario y estado. El sistema confirmó al Administrador que la operación fue exitosa y la nueva configuración queda lista para el envío automático.

Fuente: Elaboración propia

2.5.13 Caso de uso: CU13 – Administrar Usuarios

Tabla 2.28. Caso de uso narrativo: CU13 – Administrar Usuarios

Caso de uso:	Administrar Usuarios
Código:	CU13
Actor:	Profesional
Descripción:	Permite al Administrador crear, modificar, bloquear, reactivar y eliminar cuentas de usuarios.
Referencias cruzadas:	No tiene.
Precondición:	El Administrador debe haber iniciado sesión y contar con privilegios para acceder a la sección de gestión de usuarios.
Postcondición:	Los cambios realizados se guardan en el sistema y quedan operativos para sus futuros accesos o restricciones en la plataforma.
Flujo Principal	
Actor	Sistema
1. El Administrador accede a la sección “Administrar Usuarios” en la plataforma.	
	2. El sistema muestra la lista de usuarios existentes, indicando su estado.
3. El Administrador selecciona la operación que desea realizar.	
	4. El sistema despliega el formulario o la pantalla correspondiente a la operación elegida.
5. El Administrador completa o modifica los datos del usuario y confirma la acción.	
	6. El sistema valida la información ingresada y aplica los cambios en la base de datos.
	7. El sistema confirma al Administrador que la operación se realizó con éxito.

Fuente: Elaboración propia

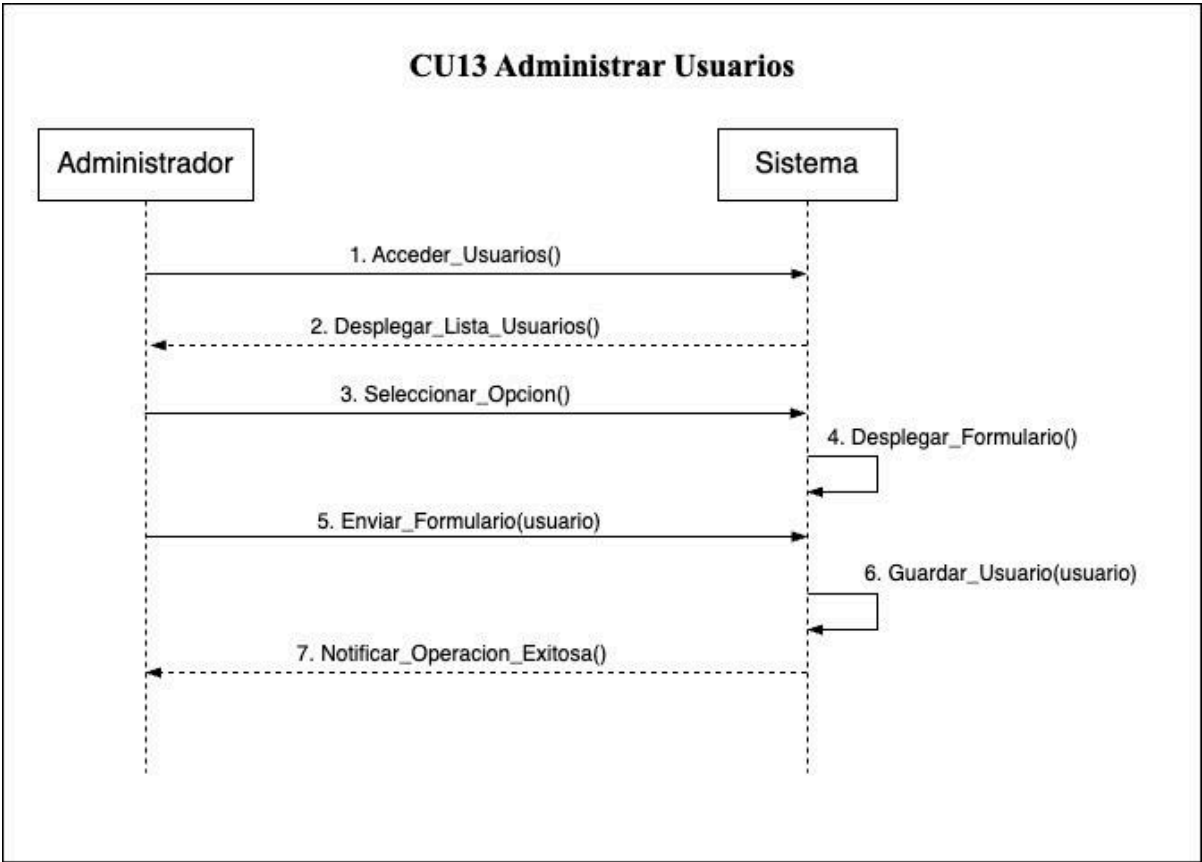


Figura 2.16. Diagrama de secuencia: CU13 – Administrar Usuarios
Fuente: Elaboración propia

Tabla 2.29. Contrato: CU13 – Administrar Usuarios

Nombre	Guardar_Usuario(usuario)
Ref. Cruzada	No tiene.
Precondición	El Administrador completó el formulario con datos del usuario y posee privilegios de administración para esta acción.
Postcondición	El usuario quedó creado/actualizado en la base de datos, el sistema confirmó la operación exitosa y muestra los cambios inmediatamente.

Fuente: Elaboración propia

2.5.14 Caso de uso: CU14 – Gestionar Conflictos

Tabla 2.30. Caso de uso narrativo: CU14 – Gestionar Conflictos

Caso de uso:	Gestionar Conflictos
Código:	CU14
Actor:	Administrador
Descripción:	Permite al Administrador recibir, revisar y resolver conflictos reportados por Clientes o Profesionales, relacionados con solicitudes, servicios o cualquier problema surgido en la plataforma. El Administrador puede ver el detalle del conflicto, comunicarse con las partes involucradas y tomar medidas.
Referencias cruzadas:	No tiene.
Precondición:	Debe haber al menos un conflicto abierto o en “Revisión” en el sistema, y el Administrador debe tener privilegios para acceder al módulo de gestión de conflictos.
Postcondición:	El estado del conflicto puede pasar a “Resuelto”. El sistema registra la resolución y/o las acciones tomadas, quedando constancia para un futuro histórico.
Flujo Principal	
Actor	Sistema
1. El Administrador accede a la sección “Gestionar Conflictos” en la plataforma.	
	2. El sistema muestra la lista de conflictos reportados, con su estado actual.
3. El Administrador selecciona un conflicto para revisarlo en detalle.	
	4. El sistema despliega la información completa del conflicto.
5. El Administrador añade notas o actualizaciones (por ejemplo, solicita información adicional a alguna parte involucrada).	
	6. El sistema registra esos cambios, manteniendo un historial de acciones, y permite al Administrador decidir cómo proceder.
	7. El sistema confirma al Administrador que la actualización fue aplicada.

Fuente: Elaboración propia

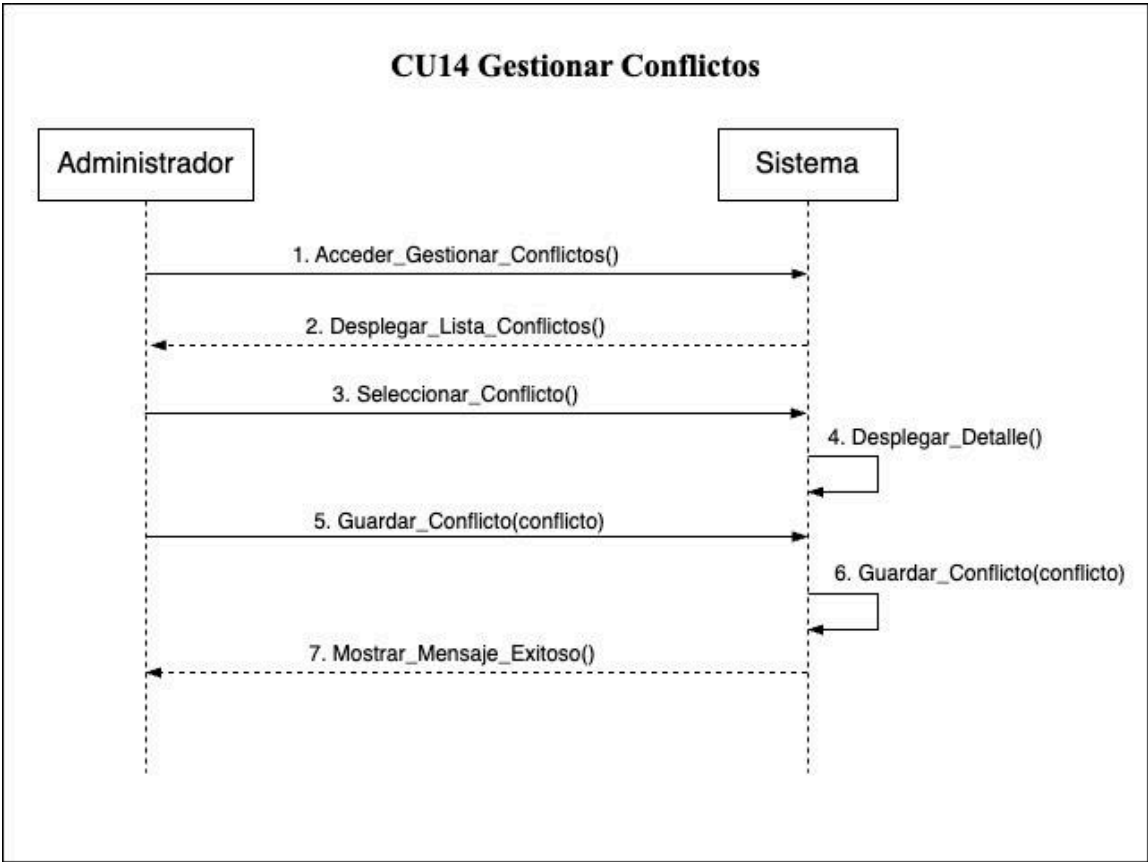


Figura 2.17. Diagrama de secuencia: CU14 – Gestionar Conflictos
Fuente: Elaboración propia

Tabla 2.31. Contrato: CU14 – Gestionar Conflictos

Nombre	Guardar_Conflicto(conflicto)
Ref. Cruzada	No tiene.
Precondición	El Administrador ha seleccionado un conflicto para revisarlo, vio sus detalles y añadió notas o actualizaciones. Además, posee privilegios para modificar conflictos.
Postcondición	Se registró la información cambiada o añadida en el conflicto, quedando un historial de acciones y permitiendo al Administrador cerrar, escalar o resolverlo más adelante. El sistema confirmó la actualización exitosa.

Fuente: Elaboración propia

2.5.15 Caso de uso: CU15 – Administrar Solicitudes

Tabla 2.32. Caso de uso narrativo: CU15 – Administrar Solicitudes

Caso de uso:	Administrar Solicitudes
Código:	CU15
Actor:	Administrador
Descripción:	El sistema permite al Administrador revisar y, en caso necesario, modificar, suspender o cancelar solicitudes que hayan sido creadas por los clientes, ya sea porque incumplan alguna normativa o se requiera una intervención administrativa.
Referencias cruzadas:	No tiene.
Precondición:	El Administrador debe haber iniciado sesión y tener privilegios de administración. Además, debe haber al menos una solicitud existente en el sistema para gestionar.
Postcondición:	La solicitud es alterada por el Administrador y el sistema registra la intervención.
Flujo Principal	
Actor	Sistema
1. El Administrador accede a la sección “Administrar Solicitudes” en la plataforma.	
	2. El sistema muestra la lista de solicitudes existentes, indicando su estado.
3. El Administrador selecciona una solicitud para revisarla.	
	4. El sistema despliega la información detallada de la solicitud.
5. El Administrador edita los datos o aplica una acción sobre la solicitud.	
	6. El sistema actualiza la solicitud con los nuevos datos.
	7. El sistema confirma al Administrador que la acción se llevó a cabo exitosamente.

Fuente: Elaboración propia

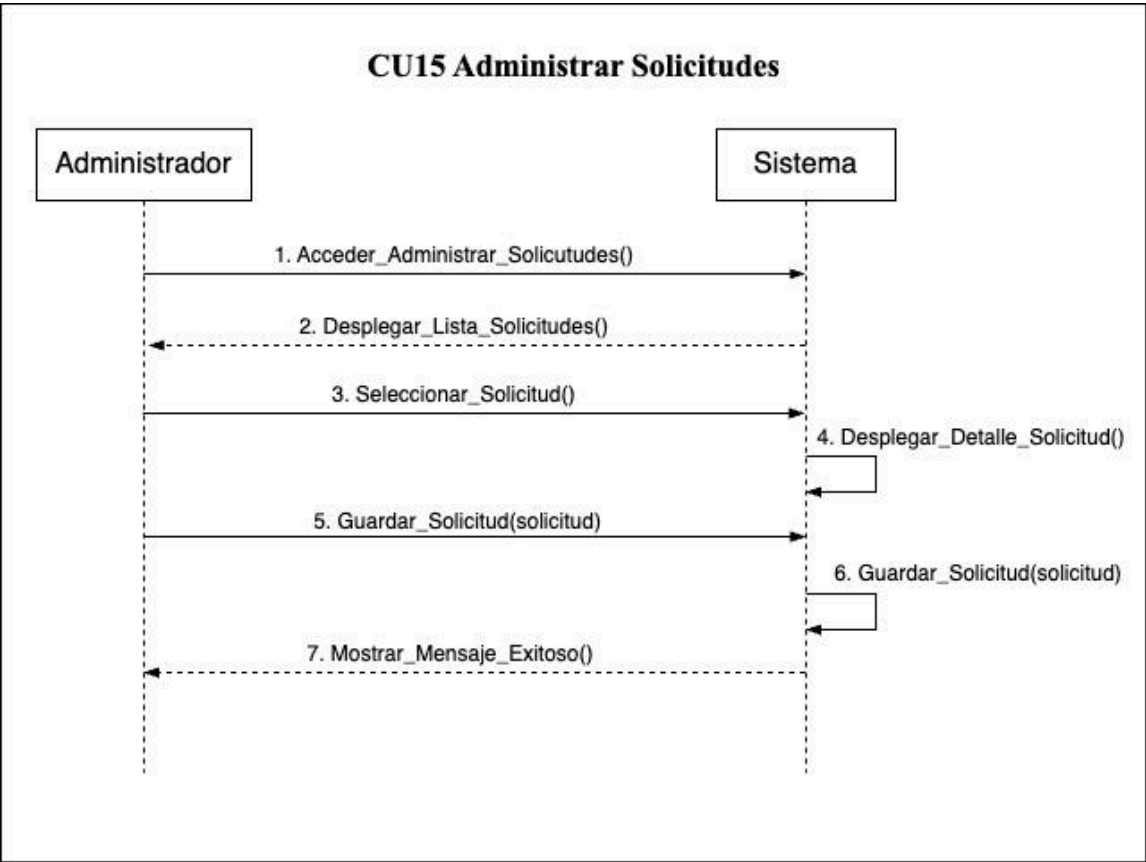


Figura 2.18. Diagrama de secuencia: CU15 – Administrar Solicitudes
Fuente: Elaboración propia

Tabla 2.33. Contrato: CU15 – Administrar Solicitudes

Nombre	Guardar_Solicitud(solicitud)
Ref. Cruzada	No tiene.
Precondición	El Administrador accedió a la sección “Administrar Solicitudes”, visualizó el detalle de la solicitud y editó o aplicó una acción. El sistema cuenta con los datos necesarios para actualizar la solicitud.
Postcondición	La solicitud quedó actualizada con los cambios realizados y el sistema registró esta intervención administrativa. Se notificó al Administrador que la operación fue exitosa.

Fuente: Elaboración propia

2.5.16 Caso de uso: CU16 – Generar Reporte

Tabla 2.34. Caso de uso narrativo: CU16 – Generar Reporte

Caso de uso:	Generar Reporte
Código:	CU16
Actor:	Administrador
Descripción:	El sistema permite al Administrador obtener reportes acerca de la actividad de la plataforma (por ejemplo, cantidad de solicitudes, servicios finalizados, calificaciones promedio, etc.). El Administrador puede seleccionar parámetros y el sistema genera un documento descargable.
Referencias cruzadas:	No tiene.
Precondición:	El Administrador debe tener iniciada la sesión y poseer privilegios para acceder al módulo de reportes. Además, debe existir información en la base de datos para generar dichos reportes.
Postcondición:	El reporte solicitado se genera.
Flujo Principal	
Actor	Sistema
1. El Administrador accede a la sección “Generar Reporte” en la plataforma.	
	2. El sistema muestra las distintas opciones de reporte.
3. El Administrador selecciona el tipo de reporte deseado y opcionalmente parámetros.	
	4. Muestra los detalles del reporte seleccionado y la opción “Descargar”.
5. El Administrador selecciona la opción “Descargar”.	
	6. El sistema descarga el archivo y muestra en pantalla que la descarga fue exitosa.

Fuente: Elaboración propia

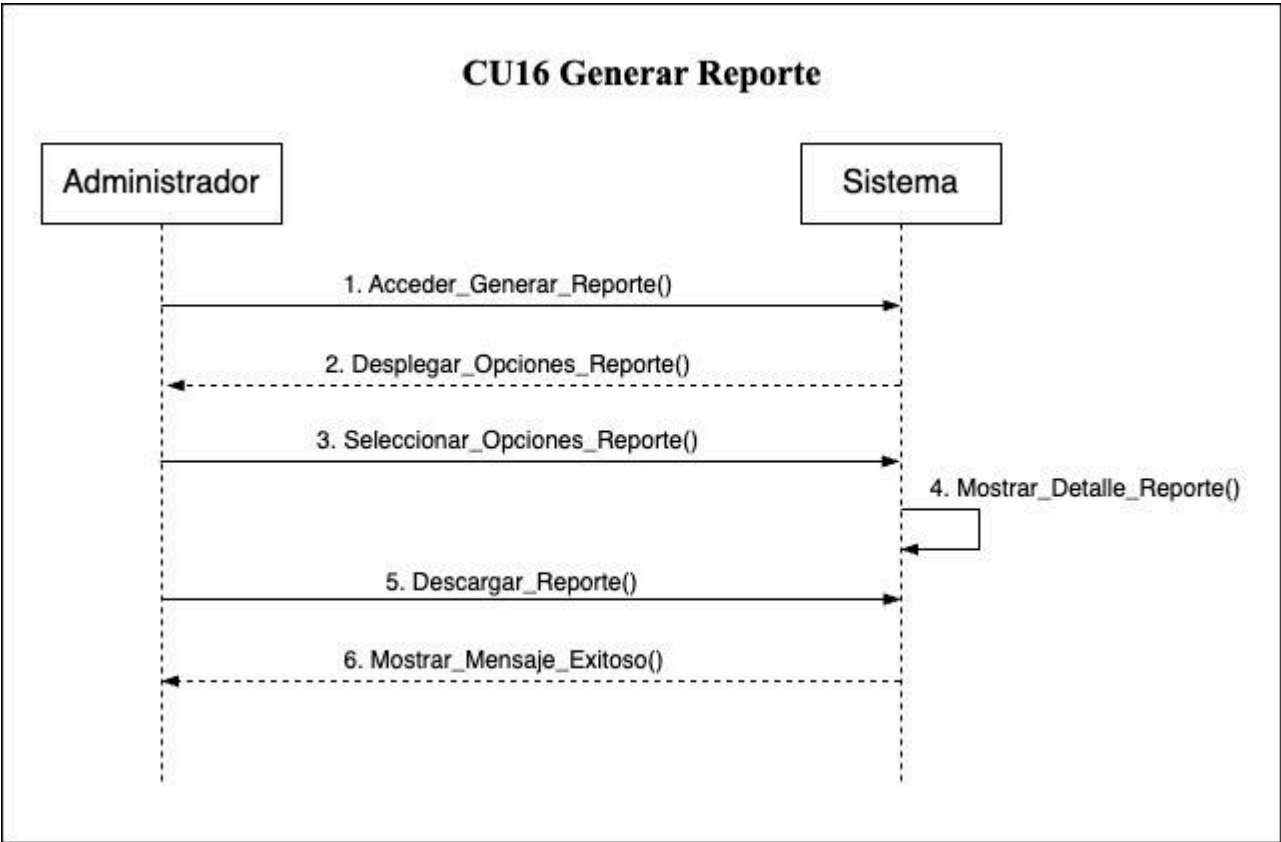


Figura 2.19. Diagrama de secuencia: CU16 – Generar Reporte
Fuente: Elaboración propia

Tabla 2.35. Contrato: CU16 – Generar Reporte

Nombre	Descargar_Reporte()
Ref. Cruzada	No tiene.
Precondición	El Administrador se encuentra autenticado, ha ingresado a la sección “Generar Reporte” y ha seleccionado el tipo de reporte, junto con sus parámetros. Además, el sistema mostró la opción “Descargar” para el archivo a generar.
Postcondición	Se generó el archivo del reporte solicitado en el formato correspondiente y se inició la descarga. El sistema confirmó que la operación se realizó correctamente, quedando el reporte listo para su uso.

Fuente: Elaboración propia

**CAPÍTULO 3. ASPECTOS RELEVANTES DEL DISEÑO PARA LA
ALTERNATIVA SELECCIONADA**

3. ASPECTOS RELEVANTES DEL DISEÑO.

En este capítulo se presenta la arquitectura del software utilizando UML, el diseño de datos a través de modelos de clases y el modelo de datos relacional. Se describen las tablas con sus respectivos campos, destacando las llaves primarias y foráneas. Además, se incluye el diseño de componentes junto con sus diagramas de colaboración correspondientes. Finalmente, se detalla el diseño de las interfaces del aplicativo, apoyado por las heurísticas de usabilidad de Jakob Nielsen.

3.1. Arquitectura del software.

Lo se propone es una aplicación web responsiva diseñada para gestionar eficientemente las solicitudes de reparación y mantenimiento del hogar. Esta solución está pensada para proporcionar a los propietarios diversas alternativas de solución con distintas cotizaciones, permitiendo así una comparación y selección informada de los profesionales adecuados.

Para desarrollar este sistema, se opta por una arquitectura en capas o también denominada N-Tier Architecture la cual ofrece una arquitectura que ofrece una separación clara de responsabilidades y facilita el mantenimiento y escalabilidad del software. Esta arquitectura incluye cuatro capas principales:

- Presentación
- Lógica de negocio
- Persistencia e Integración

3.1.1. Arquitectura del sistema

La capa de presentación es la encargada de la interacción con el usuario. Está diseñada para ser responsiva y así adaptarse a diferentes dispositivos como PC, tabletas y smartphones. Para lograrlo, se utiliza Angular versión 16.2.1 junto con Bootstrap y el módulo HttpClientModule para realizar solicitudes HTTP entre ambientes.. Este se comunicará a través del servicio REST con en el servidor de aplicaciones IIS, la capa de negocio, y la capa de datos .NET Core Vrs. 8 con web API, además de contar con autenticación y autorización para la seguridad del sistema utilizando tecnologías como JWT (JSON Web Tokens) y ASP.NET Identity, integradas con una base de datos PostgreSQL versión 16.2.

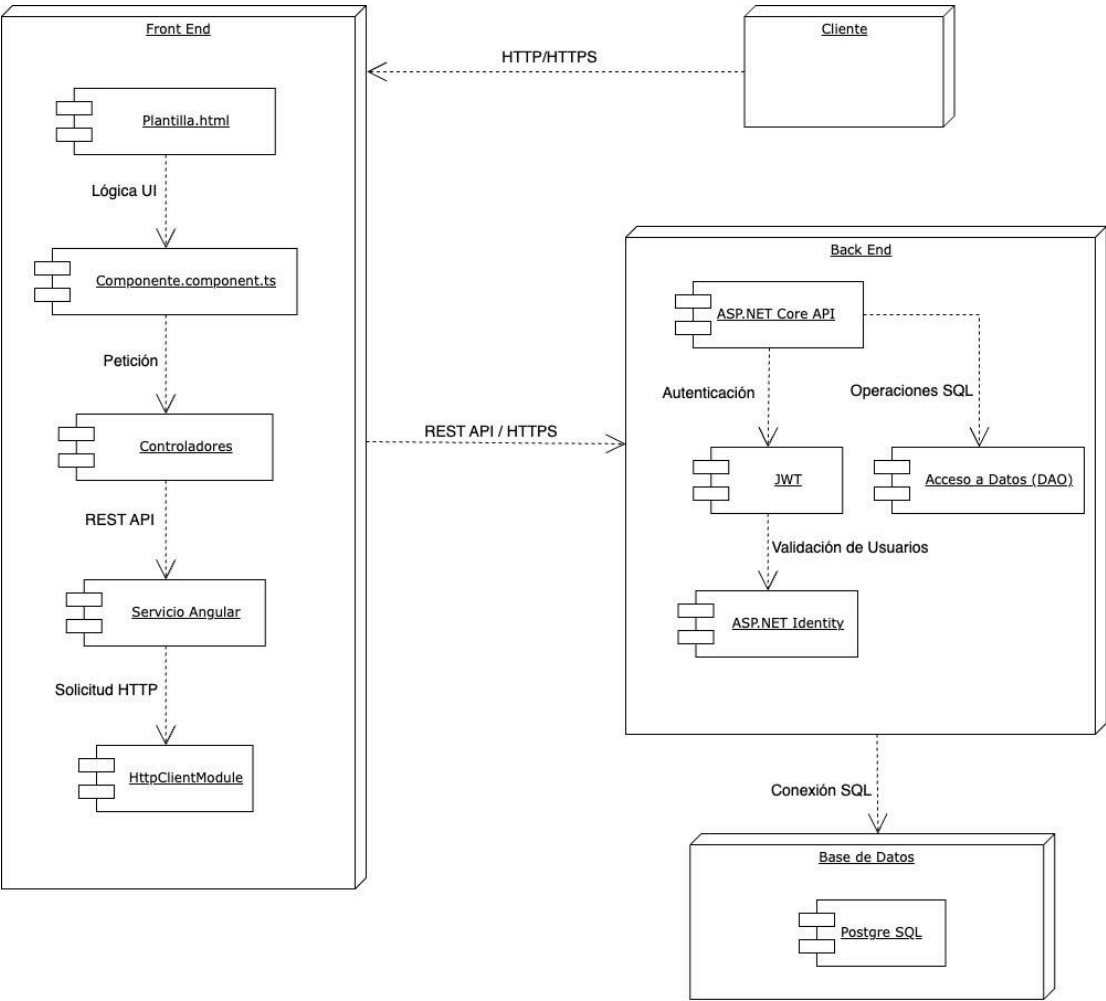


Figura 3.1. Arquitectura del sistema
Fuente: Elaboración propia

3.1.2. Características del recurso computacional

Para cada desarrollador se le entregará un equipo de trabajo con las siguientes especificaciones:

- Sistema operativo: Microsoft Windows 11 Pro de 64 bits.
- Procesador: Intel® Core™ i9 -13980HX CPU @ 5.6GHz.
- Memoria RAM: 32 GB de RAM.

Al igual que ocuparán lenguajes de programación y herramientas como:

- Herramienta UML para la elaboración de diagramas
- C# .NET Core lenguaje para backend
- PostgreSQL para el manejo de datos.
- Visual Studio 2022 para desarrollar el backend.
- Visual Studio Code para desarrollar el frontend.
- Git y GitHub para control de versiones.
- Postman para pruebas de endpoints.
- xUnit y NUnit Framework para pruebas unitarias.

Por último, para el Framework se ocupará:

- Angular Vrs. 16.2.1

3.2. Diseño de datos

En esta sección se describe el diagrama de clases y el modelo relacional, destacando las claves primarias y foráneas de cada tabla del modelo. En esta etapa se presentarán los modelos de clase, y posteriormente el modelo relacional del sistema, para terminar con el diccionario de datos, que corresponde a la descripción de atributos de las tablas obtenidas en el modelo relacional.

3.2.1. Modelo de clases

En la figura 3.2 se presenta el modelo de clases, obtenido a partir del modelo conceptual visto en el capítulo.

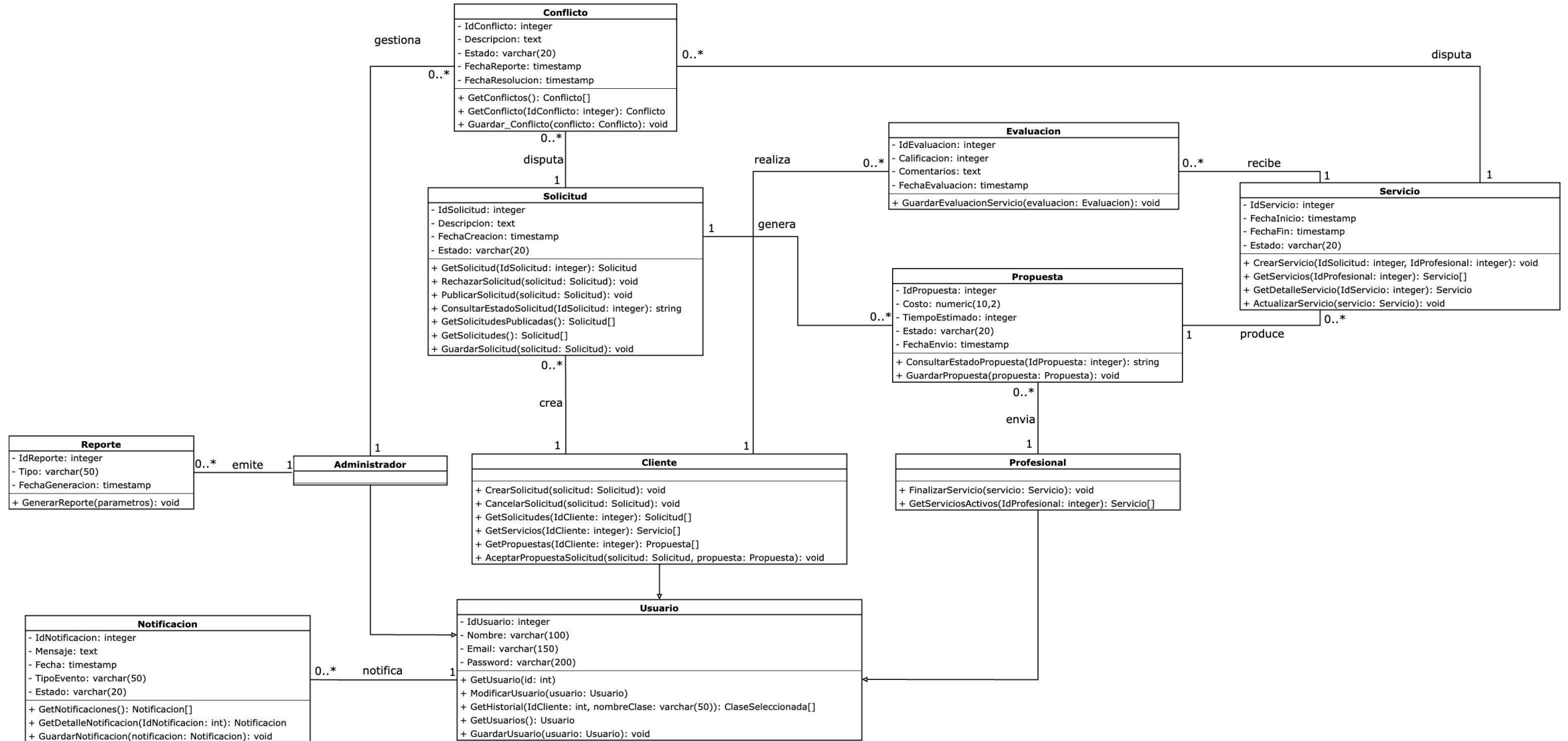


Figura 3.2. Modelo de clases

Fuente: Elaboración propia

3.2.2. Modelo relacional

En la figura 3.3, se presenta el modelo relacional del sistema, donde se muestra en detalle, tanto las llaves primarias (PK) como las llaves secundarias (FK) de las tablas que se relacionan.

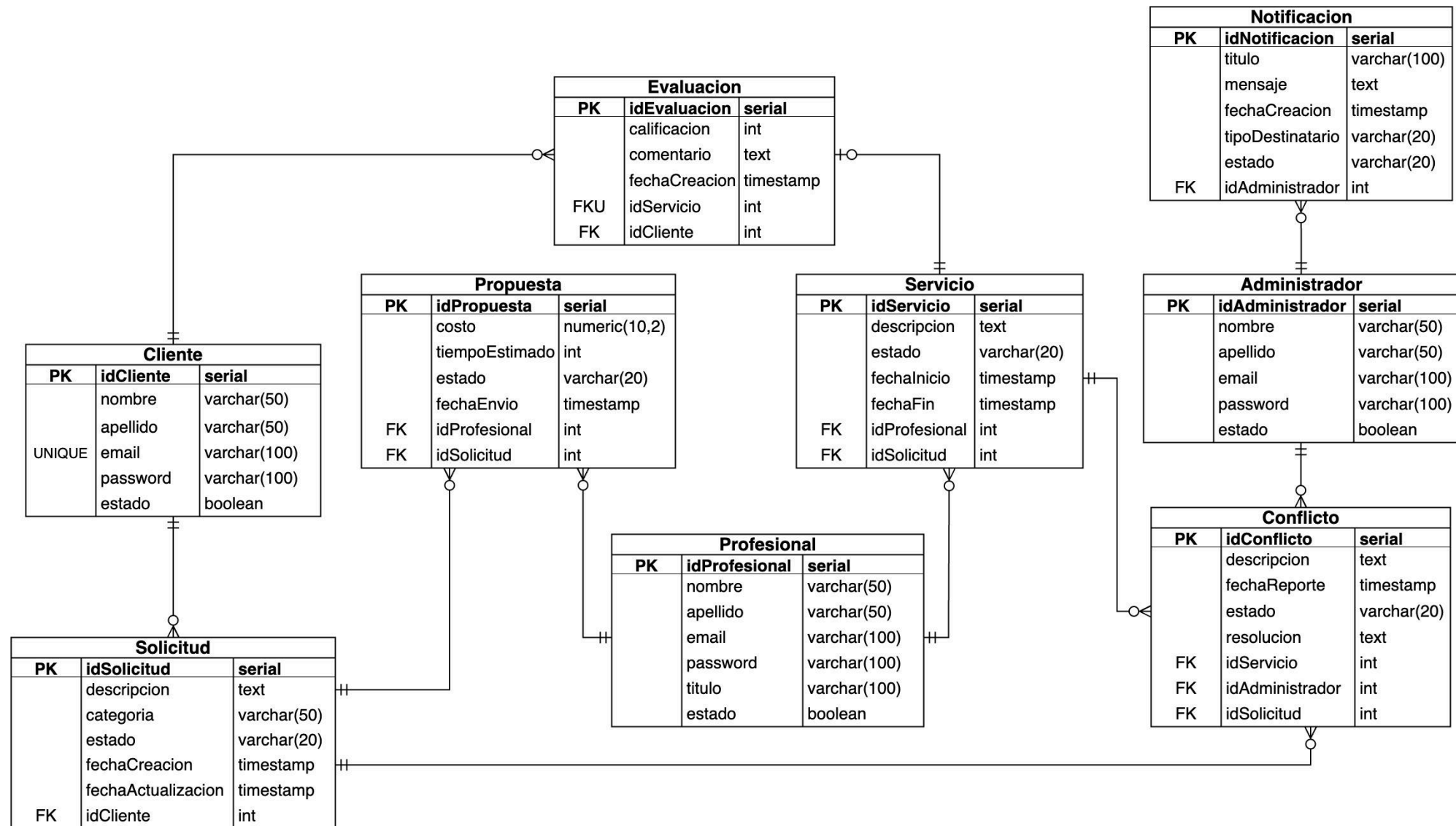


Figura 3.3. Modelo Relacional

Fuente: Elaboración prop

3.2.3. Diccionario de datos

En esta sección se describe en detalle las tablas generadas para el sistema, en las cuales destaca la representación de las llaves: PK para la llave primaria y FK para la llave foránea.

Tabla 3.1. Diccionario de datos: Administrador

Nombre Tabla	Administrador	
Descripción	Esta tabla contiene los administradores del sistema, quienes supervisan y monitorean entidades clave como solicitudes y servicios.	
Clave Primaria	idAdministrador	
Clave foránea	Ninguna.	
Nombre Campo	Tipo	Descripción
idAdministrador	serial	Identificador único y autonumérico para los administradores. Es la clave primaria y no permite valores nulos.
nombre	varchar(50)	Texto libre que almacena el nombre del administrador. Es obligatorio, no permite valores nulos.
apellido	varchar(50)	Texto libre que almacena el apellido del administrador. Es obligatorio, no permite valores nulos.
email	varchar(100)	Almacena el correo electrónico del administrador. Este campo es obligatorio, único y no permite valores nulos.
password	varchar(100)	Almacena la contraseña cifrada del administrador. Es obligatorio, no permite valores nulos y se almacenará hasheada.
estado	boolean	Indica el estado del administrador (true = activo, false = inactivo). Es obligatorio, no permite valores nulos.

Fuente: Elaboración propia.

Tabla 3.2. Diccionario de datos tabla: Cliente

Nombre Tabla	Cliente	
Descripción	Esta tabla contiene los clientes del sistema, quienes pueden crear solicitudes y calificar servicios.	
Clave Primaria	idCliente	
Clave foránea	Ninguna.	
Nombre Campo	Tipo	Descripción
idCliente	serial	Identificador único y autonumérico para los clientes del sistema. Es la clave primaria y no permite valores nulos.
nombre	varchar(50)	Texto libre que almacena el nombre del cliente. Es obligatorio, no permite valores nulos.
apellido	varchar(50)	Texto libre que almacena el apellido del cliente. Es obligatorio, no permite valores nulos.
email	varchar(100) UNIQUE	Correo electrónico del cliente. Es obligatorio, no permite valores nulos y debe ser único en la tabla (restricción UNIQUE).
password	varchar(100)	Almacena la contraseña cifrada del cliente. Es obligatorio, no permite valores nulos y se almacenará hasheada
estado	boolean	Indica el estado del cliente (true = activo, false = inactivo). Es obligatorio, no permite valores nulos.

Fuente: Elaboración propia

Tabla 3.3. Diccionario de datos tabla: Solicitud

Nombre Tabla	Solicitud	
Descripción	Esta tabla registra las solicitudes creadas por los clientes para reparar o mantener servicios.	
Clave Primaria	idSolicitud	
Clave foránea	idCliente → Cliente.idCliente	
Nombre Campo	Tipo	Descripción
idSolicitud	serial	Identificador único y autonumérico para las solicitudes. Es la clave primaria y no permite valores nulos.
descripcion	text	Texto libre que describe detalladamente el problema. Es obligatorio y no permite valores nulos.
categoria	varchar(50)	Categoría del problema (por ejemplo, plomería, electricidad). Es obligatorio y no permite valores nulos.
estado	varchar(20)	Estado actual de la solicitud. Es obligatorio y no permite valores nulos.
fechaCreacion	timestamp	Fecha y hora en la que se creó la solicitud. Es obligatorio y no permite valores nulos.
fechaActualizacion	timestamp	Fecha y hora de la última actualización de la solicitud. Es obligatorio y no permite valores nulos.
idCliente	int	Identificador del cliente que realiza la solicitud. Es una clave foránea relacionada con Cliente.idCliente.

Fuente: Elaboración propia

Tabla 3.4. Diccionario de datos tabla: Servicio

Nombre Tabla	Servicio	
Descripción	Esta tabla registra los servicios realizados por los profesionales en respuesta a solicitudes de los clientes.	
Clave Primaria	idServicio	
Clave foránea	idProfesional → Profesional.idProfesional idSolicitud → Solicitud.idSolicitud	
Nombre Campo	Tipo	Descripción
idServicio	serial	Identificador único y autonumérico para los servicios. Es la clave primaria y no permite valores nulos.
descripcion	text	Texto libre que describe detalladamente el servicio realizado. Es obligatorio y no permite valores nulos.
estado	varchar(20)	Estado actual del servicio (pendiente, completado, etc.). Es obligatorio y no permite valores nulos.
fechaInicio	timestamp	Fecha y hora en la que se inició el servicio. Es obligatorio y no permite valores nulos.
fechaFin	timestamp	Fecha y hora en la que se finalizó el servicio. Es obligatorio y no permite valores nulos.
idProfesional	int	Identificador del profesional que realiza el servicio. Es una clave foránea relacionada con Profesional.idProfesional.
idSolicitud	int	Identificador de la solicitud asociada al servicio. Es una clave foránea relacionada con Solicitud.idSolicitud.

Fuente: Elaboración propia

Tabla 3.5. Diccionario de datos tabla: Profesional

Nombre Tabla	Profesional	
Descripción	Esta tabla contiene los profesionales registrados, quienes realizan servicios en respuesta a las solicitudes de los clientes.	
Clave Primaria	idProfesional	
Clave foránea	Ninguna	
Nombre Campo	Tipo	Descripción
idProfesional	serial	Identificador único y autonumérico para los profesionales. Es la clave primaria y no permite valores nulos.
nombre	varchar(50)	Texto libre que almacena el nombre del profesional. Es obligatorio y no permite valores nulos.
apellido	varchar(50)	Texto libre que almacena el apellido del profesional. Es obligatorio y no permite valores nulos.
email	varchar(100)	Correo electrónico único del profesional. Es obligatorio, no permite valores nulos.
password	varchar(100)	Almacena la contraseña cifrada del profesional. Es obligatorio, no permite valores nulos y se almacenará hasheada
titulo	varchar(100)	Texto que indica la especialización o título del profesional. Es obligatorio y no permite valores nulos.
estado	varchar(20)	Estado de la cuenta del profesional. No admite valores nulos.

Fuente: Elaboración propia

Tabla 3.6. Diccionario de datos tabla: Evaluación

Nombre Tabla	Evaluación	
Descripción	Esta tabla registra las evaluaciones realizadas por los clientes sobre los servicios prestados.	
Clave Primaria	idEvaluacion	
Clave foránea	idServicio → Servicio.idServicio (FKU) idCliente → Cliente.idCliente	
Nombre Campo	Tipo	Descripción
idEvaluacion	serial	Identificador único y autonumérico para las evaluaciones. Es la clave primaria y no permite valores nulos.
calificacion	int	Valor numérico que representa la calificación del servicio (1-5). Es obligatorio y no permite valores nulos.
comentario	text	Texto que almacena los comentarios del cliente sobre el servicio prestado. Es opcional.
fechaCreacion	timestamp	Fecha y hora en la que se creó la evaluación. Es obligatorio y no permite valores nulos.
idServicio	int	Identificador del servicio evaluado. Es una clave foránea única (FKU) que garantiza una relación uno a uno con Servicio.idServicio. Esto implica que un servicio puede tener, como máximo, una evaluación asociada.
idCliente	int	Identificador del cliente quien realiza la evaluación. Es una clave foránea relacionada con Cliente.idCliente.

Fuente: Elaboración propia

Tabla 3.7. Diccionario de datos tabla: Conflicto

Nombre Tabla	Conflicto	
Descripción	Esta tabla registra los conflictos reportados entre clientes y profesionales relacionados con los servicios realizados.	
Clave Primaria	idConflicto	
Clave foránea	idServicio → Servicio.idServicio idAdministrador → Administrador.idAdministrador idSolicitud → Solicitud.idSolicitud	
Nombre Campo	Tipo	Descripción
idConflicto	serial	Identificador único y autonumérico para los conflictos. Es la clave primaria y no permite valores nulos.
descripcion	text	Texto que describe el conflicto reportado. Es obligatorio y no permite valores nulos.
fechaReporte	timestamp	Fecha y hora en la que se reportó el conflicto. Es obligatorio y no permite valores nulos.
estado	varchar(20)	Estado actual del conflicto (pendiente, resuelto, etc.). Es obligatorio y no permite valores nulos.
resolucion	text	Texto que describe la resolución del conflicto, si aplica. Es opcional.
idServicio	int	Identificador del servicio asociado al conflicto. Es una clave foránea relacionada con Servicio.idServicio.
idAdministrador	int	Identificador del administrador que gestiona el conflicto. Es una clave foránea relacionada con Administrador.idAdministrador.
idSolicitud	int	Identificador de la solicitud asociado al conflicto. Es una clave foránea relacionada con Solicitud.idSolicitud.

Fuente: Elaboración propia

Tabla 3.8. Diccionario de datos tabla: Propuesta

Nombre Tabla	Propuesta	
Descripción	Almacena las propuestas que los profesionales envían a las solicitudes publicadas por los clientes, con información de costo, tiempo estimado y estado de la propuesta.	
Clave Primaria	idPropuesta	
Clave foránea	idProfesional → Profesional.idProfesional idSolicitud → Solicitud.idSolicitud	
Nombre Campo	Tipo	Descripción
idPropuesta	serial	Identificador único y autonumérico para los conflictos. Es la clave primaria y no permite valores nulos.
costo	numeric(10,2)	Costo estimado de la propuesta. No admite valores nulos.
tiempoEstimado	int	Tiempo estimado en días para completar el servicio. No admite valores nulos.
estado	varchar(20)	Estado de la propuesta (“enviada”, “aceptada”, “rechazada”). No admite valores nulos.
fechaEnvio	timestamp	Fecha y hora en que el Profesional envía la propuesta. Por defecto, la hora actual.
idProfesional	int	Identificador del profesional que realiza la propuesta. Es una clave foránea relacionada con Profesional.idProfesional.
idSolicitud	int	Identificador de la solicitud asociada a la propuesta. Es una clave foránea relacionada con Solicitud.idSolicitud.

Fuente: Elaboración propia

Tabla 3.9. Diccionario de datos tabla: Notificacion

Nombre Tabla	Notificacion	
Descripción	Almacena mensajes o avisos que el sistema envía a distintos roles cuando ocurren eventos clave. Permite al Administrador crear campañas o notificaciones específicas.	
Clave Primaria	idNotificacion	
Clave foránea	idAdministrador → Administrador.idAdministrador	
Nombre Campo	Tipo	Descripción
idNotificacion	serial	Identificador único y autonumérico para la notificación. Es la clave primaria y no permite valores nulos.
titulo	varchar(100)	Título breve de la notificación. No admite valores nulos.
mensaje	text	Contenido o cuerpo de la notificación. No admite valores nulos.
fechaCreacion	timestamp	Fecha y hora en que el Profesional envía la propuesta. Por defecto, la hora actual.
tipoDestinatario	varchar(20)	Indica a qué rol va dirigida la notificación. No admite valores nulos.
estado	varchar(20)	Indica el estado de la notificación (“vigente” e “inactiva”). Este campo es clave para determinar si la notificación se debe considerar en el sistema.
idAdministrador	int	Identificador del administrador que gestiona la notificación. Es una clave foránea relacionada con Administrador.idAdministrador.

Fuente: Elaboración propia

3.3. Diagrama de secuencia extendido

A continuación, se presentarán los diagramas de secuencia extendidos correspondientes a los principales casos de uso del sistema.

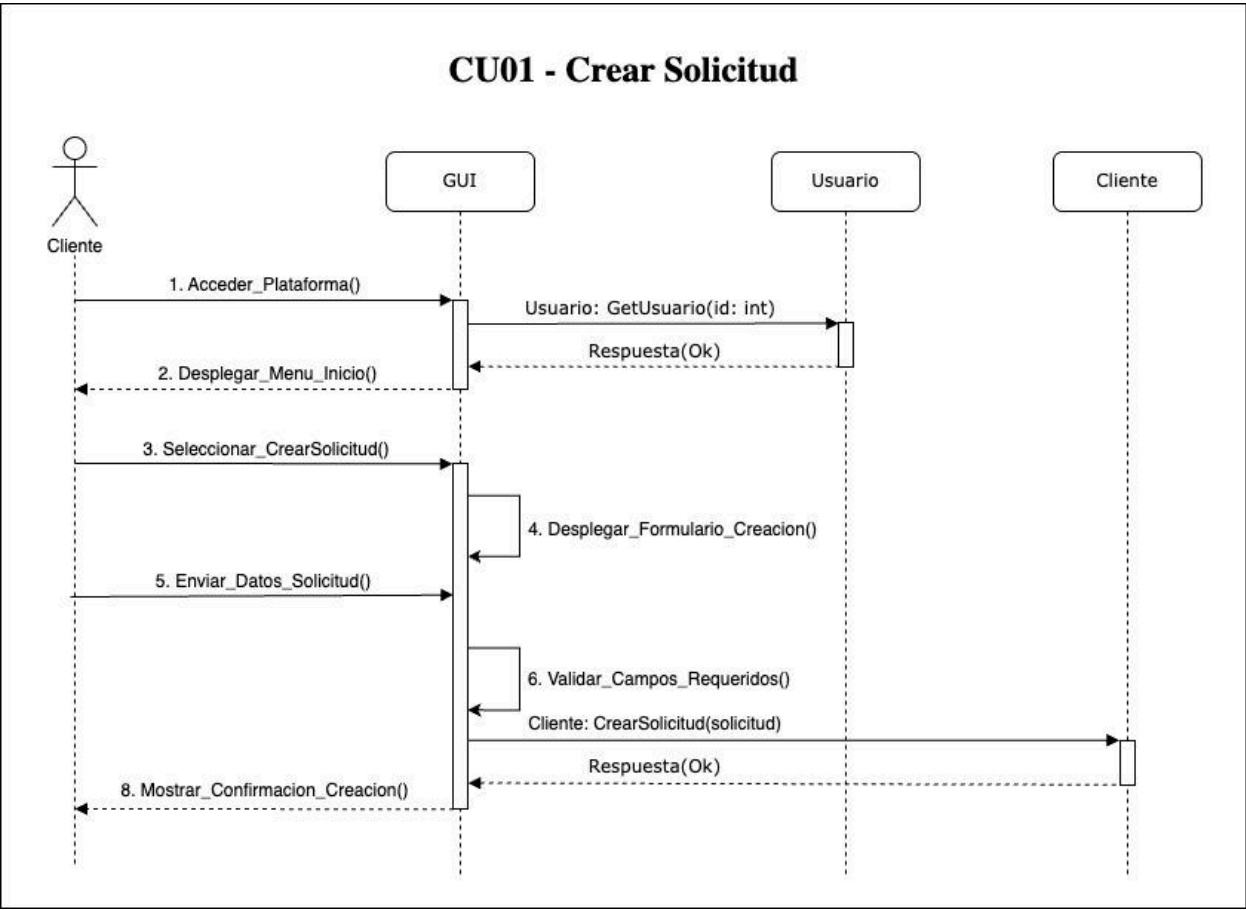


Figura 3.4. Diagrama de secuencia extendido: CU01 – Crear Solicitud
Fuente: Elaboración propia

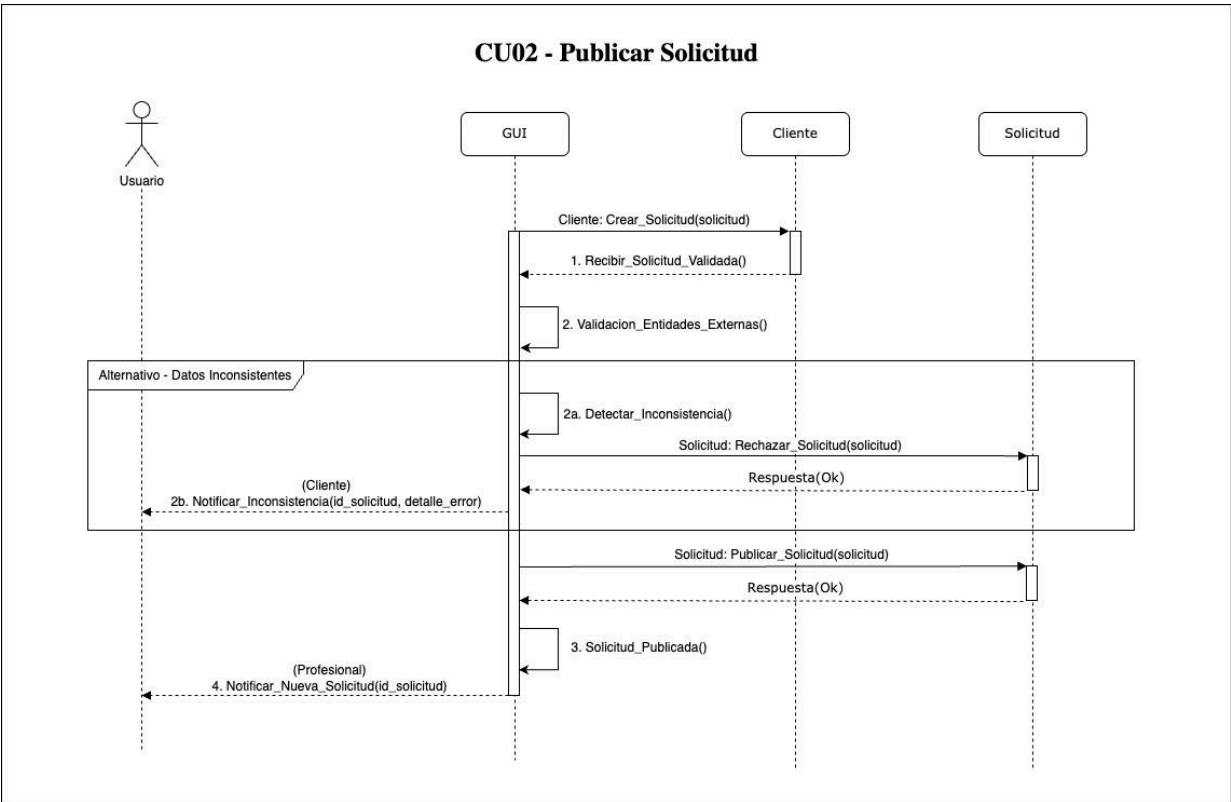


Figura 3.5. Diagrama de secuencia extendido: CU02 – Publicar Solicitud
Fuente: Elaboración propia

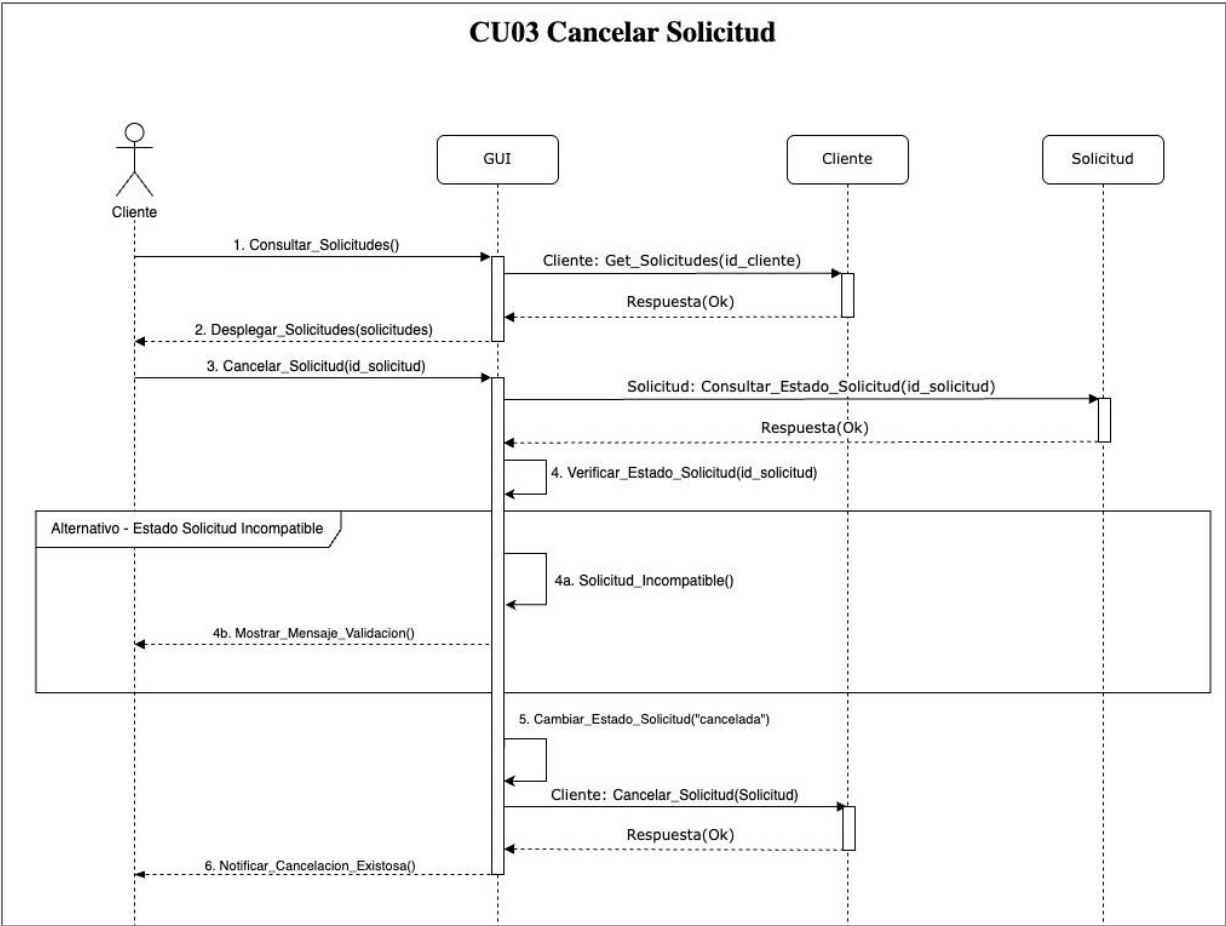


Figura 3.6. Diagrama de secuencia extendido: CU03 Cancelar Solicitud
Fuente: Elaboración propia

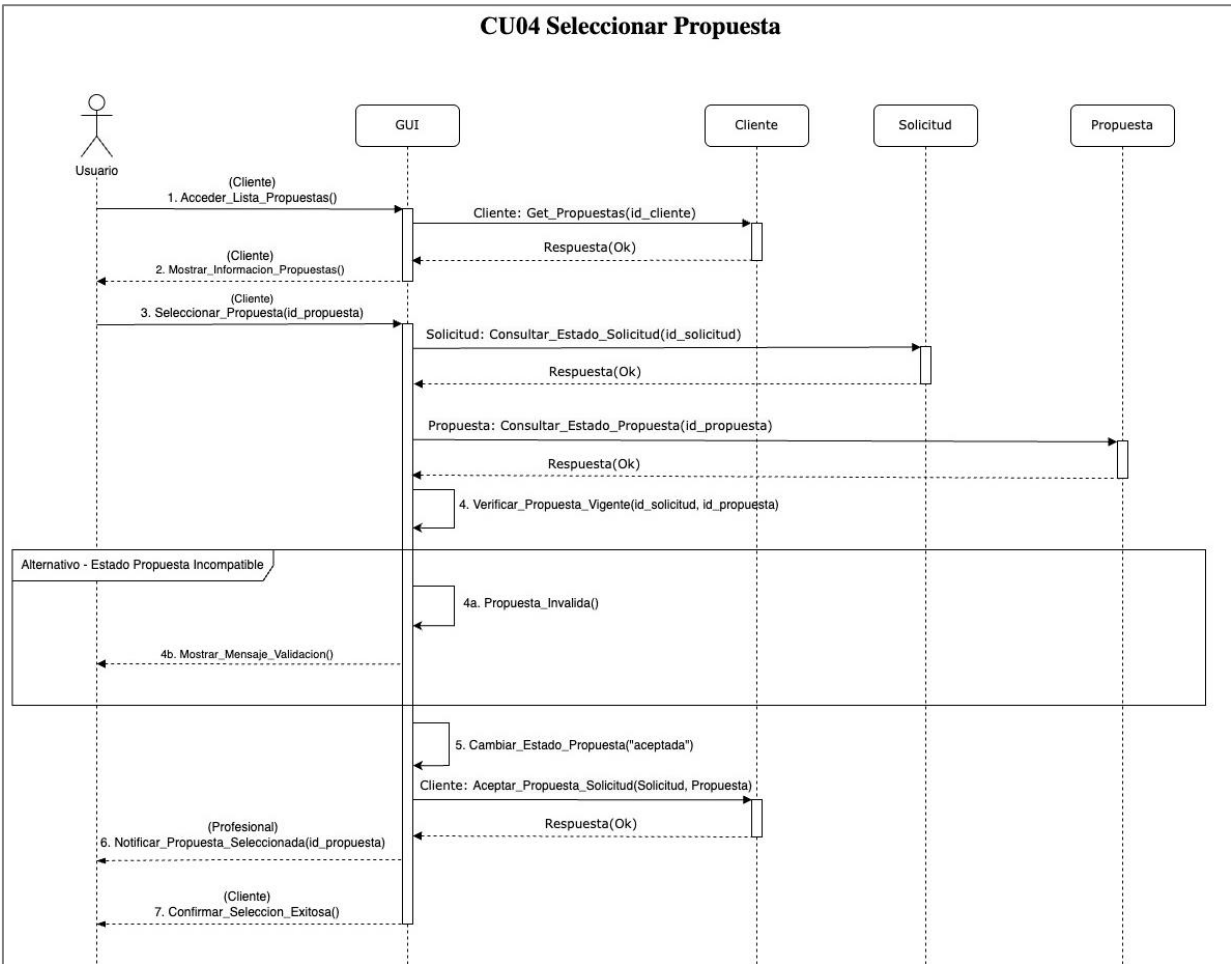


Figura 3.7. Diagrama de secuencia extendido: CU04 – Seleccionar Propuesta
Fuente: Elaboración propia

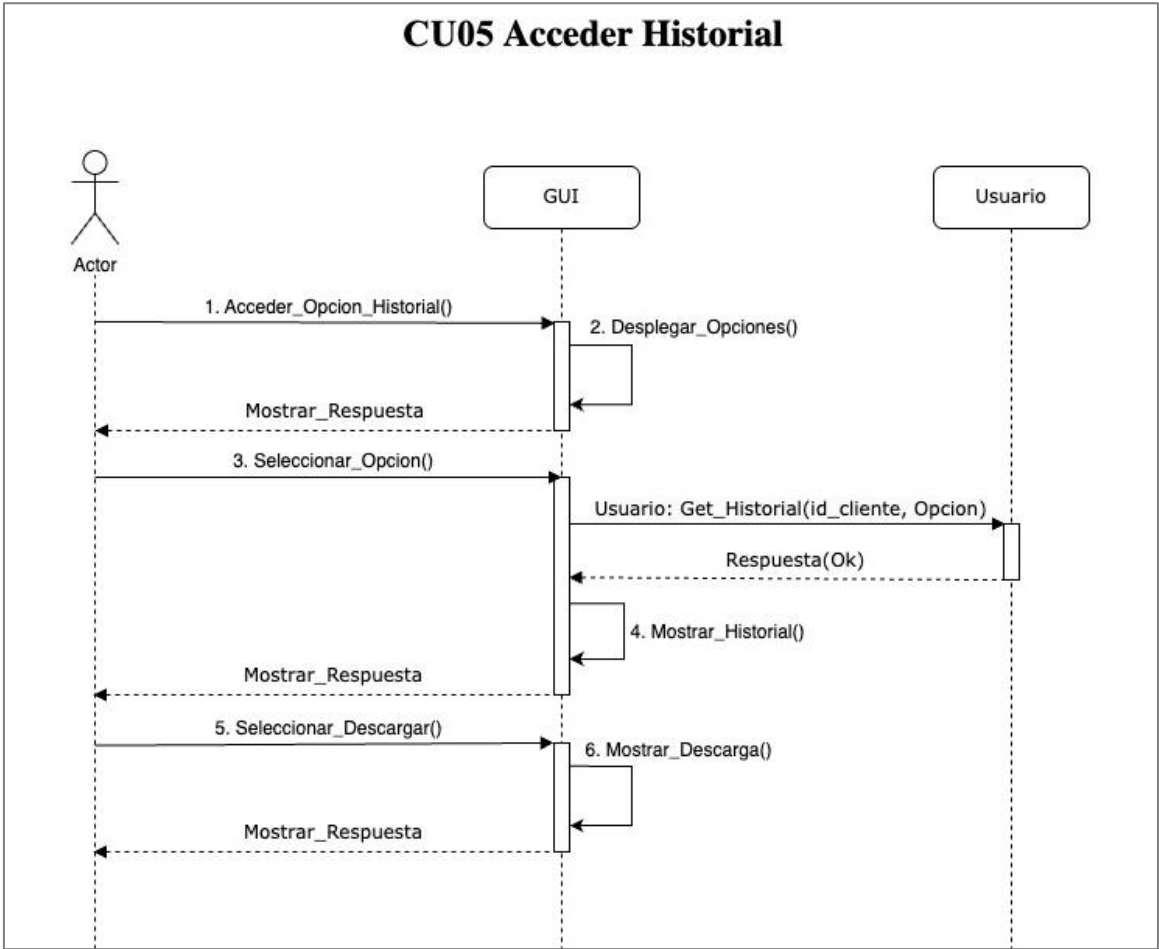


Figura 3.8. Diagrama de secuencia extendido: CU05 – Acceder Historial
Fuente: Elaboración propia

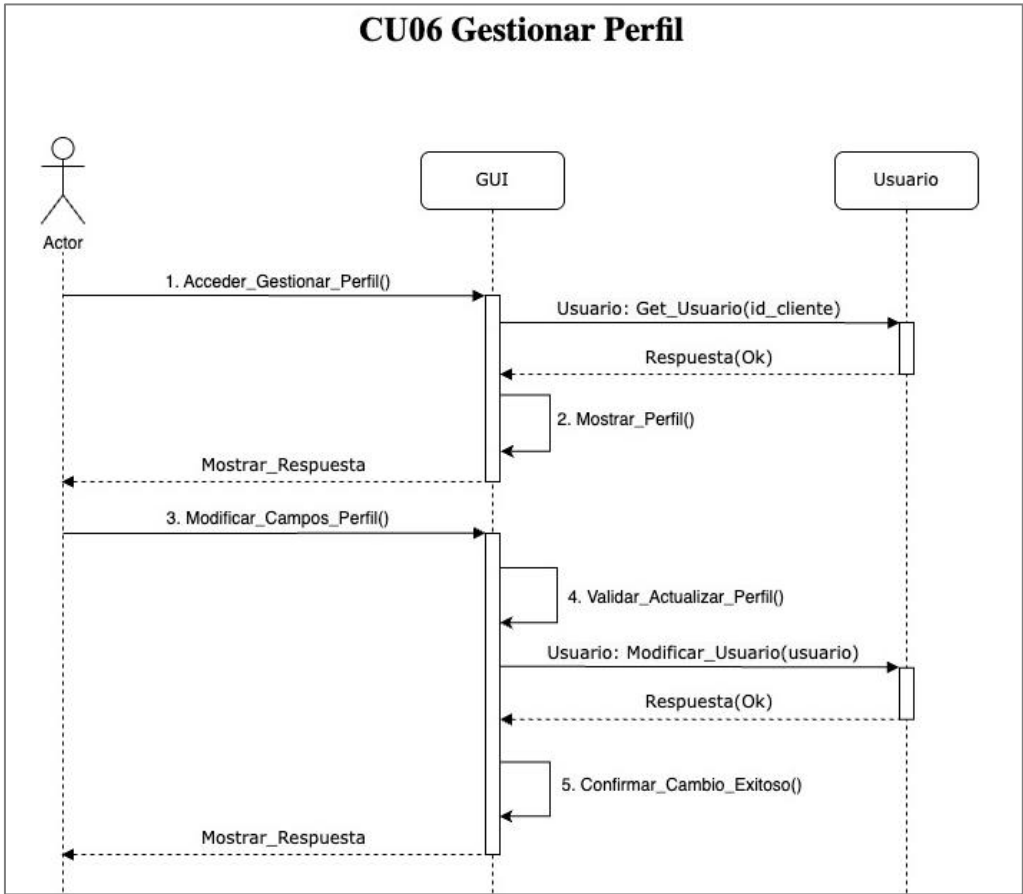


Figura 3.9. Diagrama de secuencia extendido: CU06 – Gestionar Perfil
Fuente: Elaboración propia

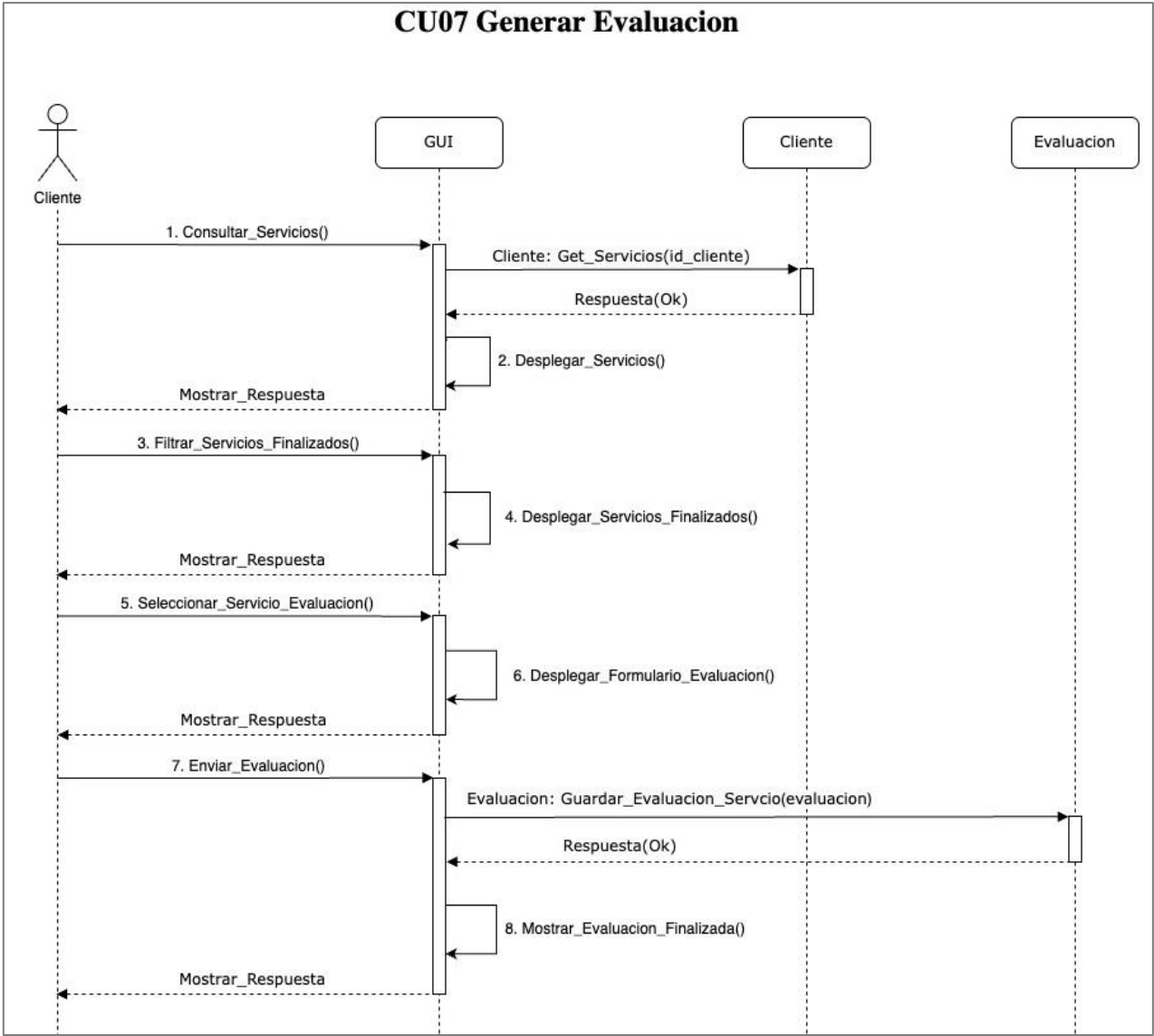


Figura 3.10. Diagrama de secuencia extendido: CU07 – Generar Evaluación
Fuente: Elaboración propia

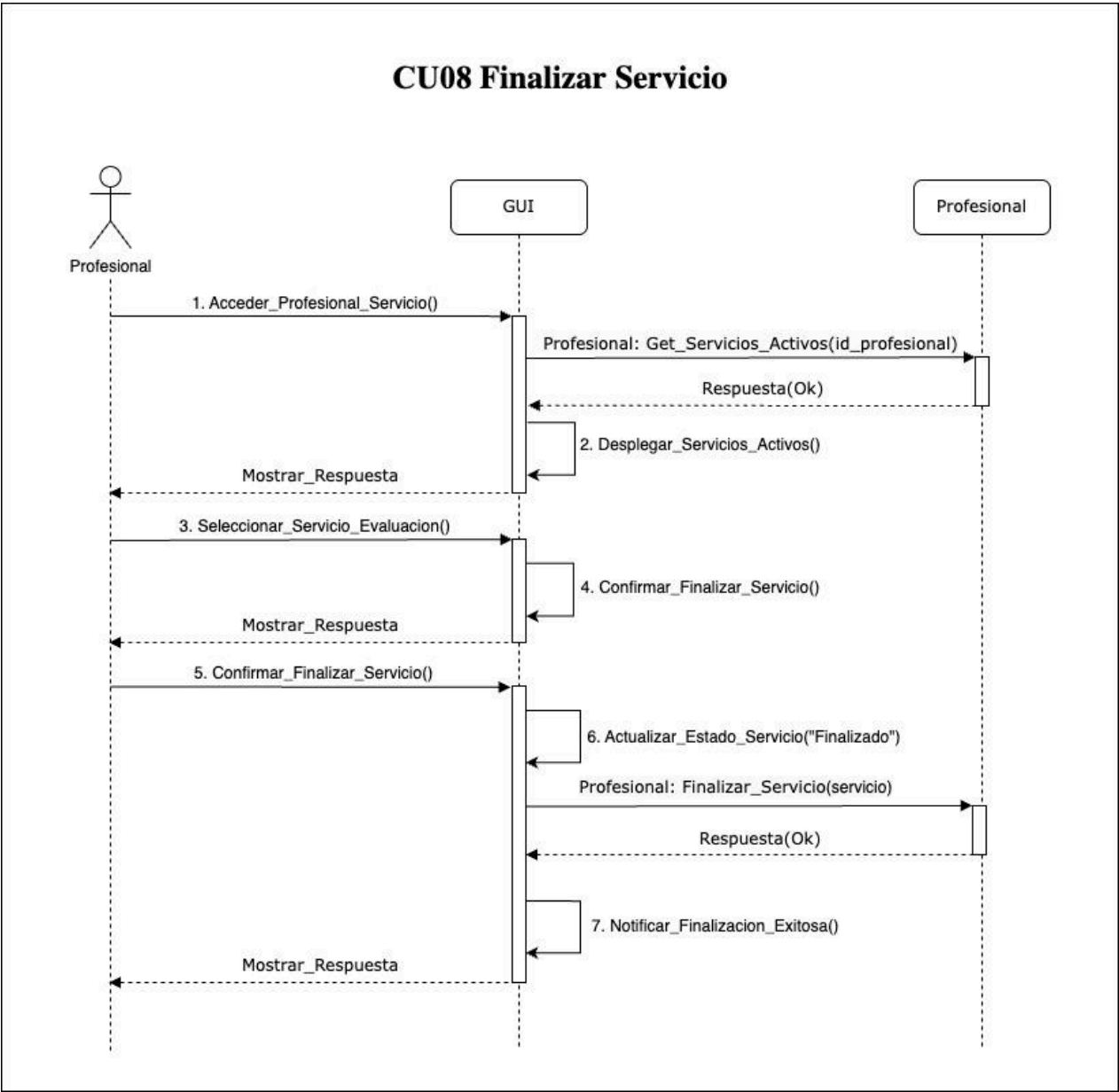


Figura 3.11. Diagrama de secuencia extendido: CU08 – Finalizar Servicio
Fuente: Elaboración propia

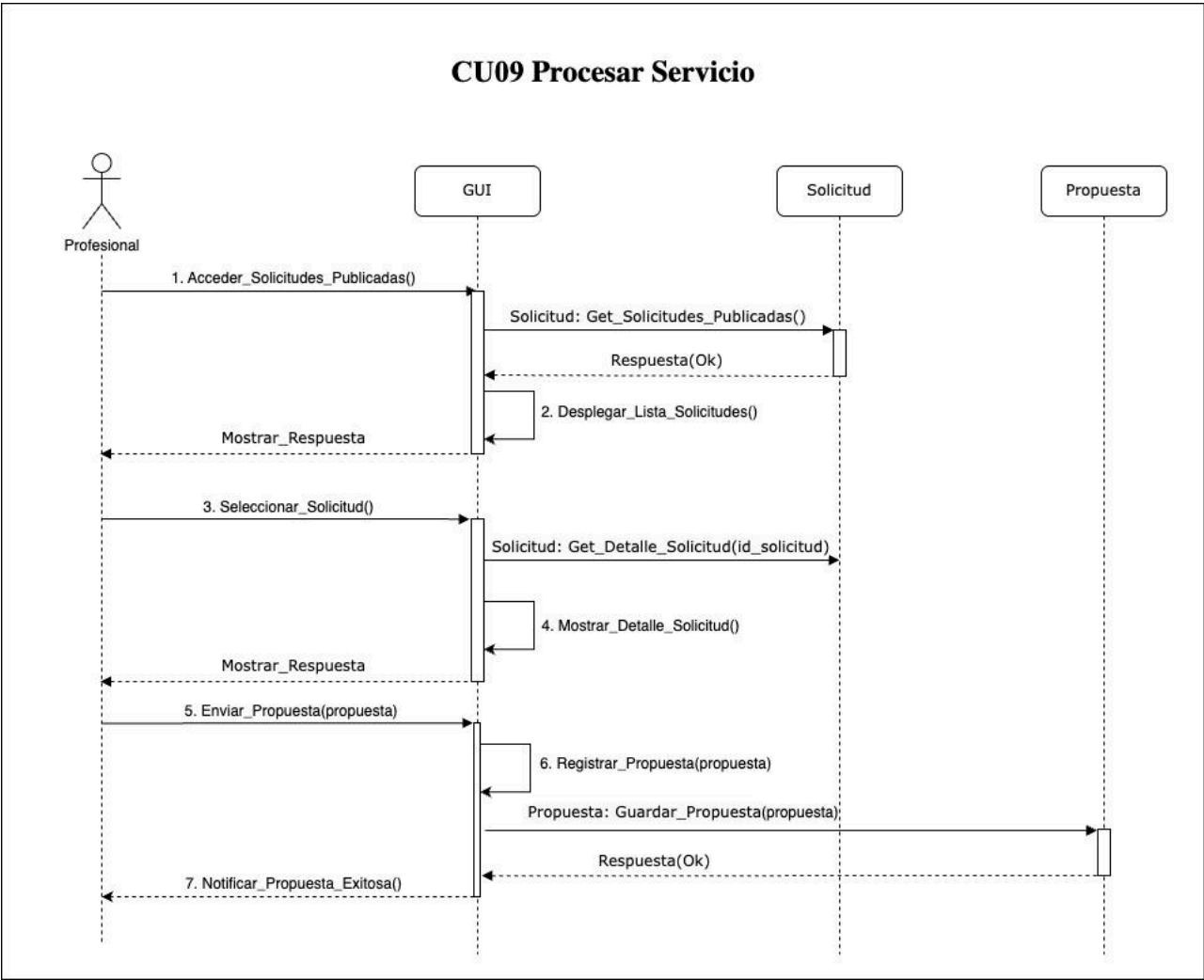


Figura 3.12. Diagrama de secuencia extendido: CU09 – Procesar Servicio
Fuente: Elaboración propia

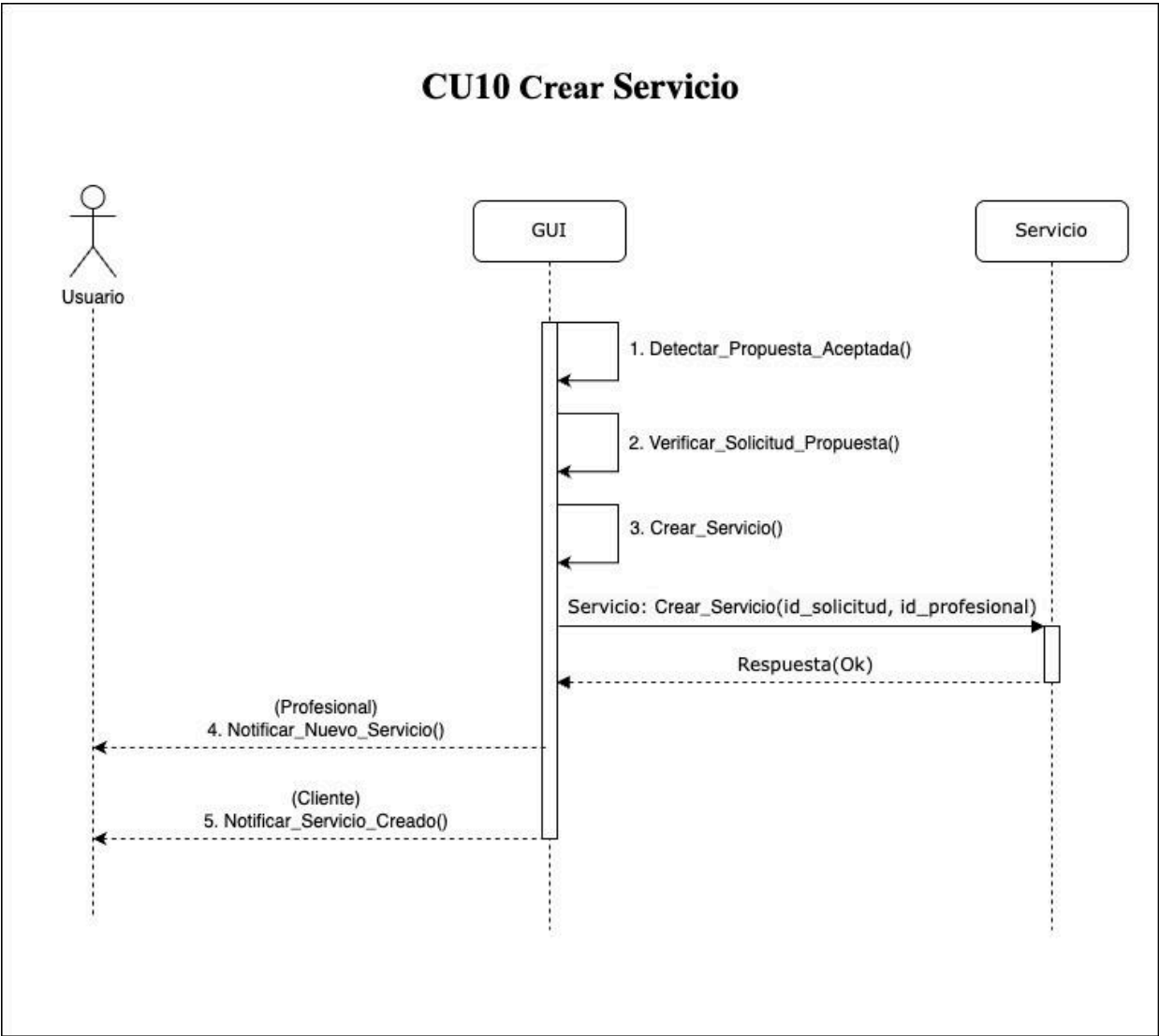


Figura 3.13. Diagrama de secuencia extendido: CU10 – Crear Servicio
Fuente: Elaboración propia

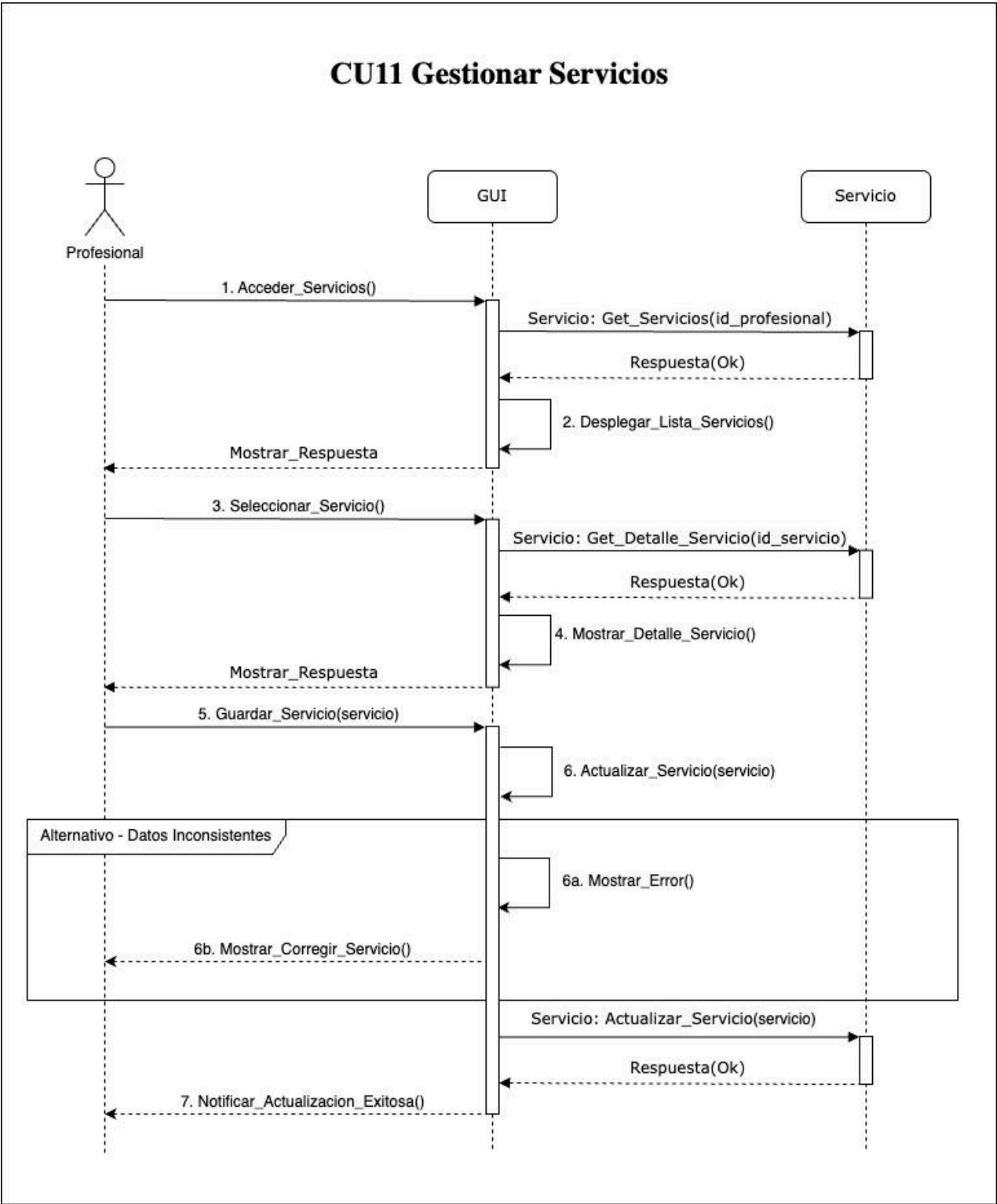


Figura 3.14. Diagrama de secuencia extendido: CU11 – Gestionar Servicio
Fuente: Elaboración propia

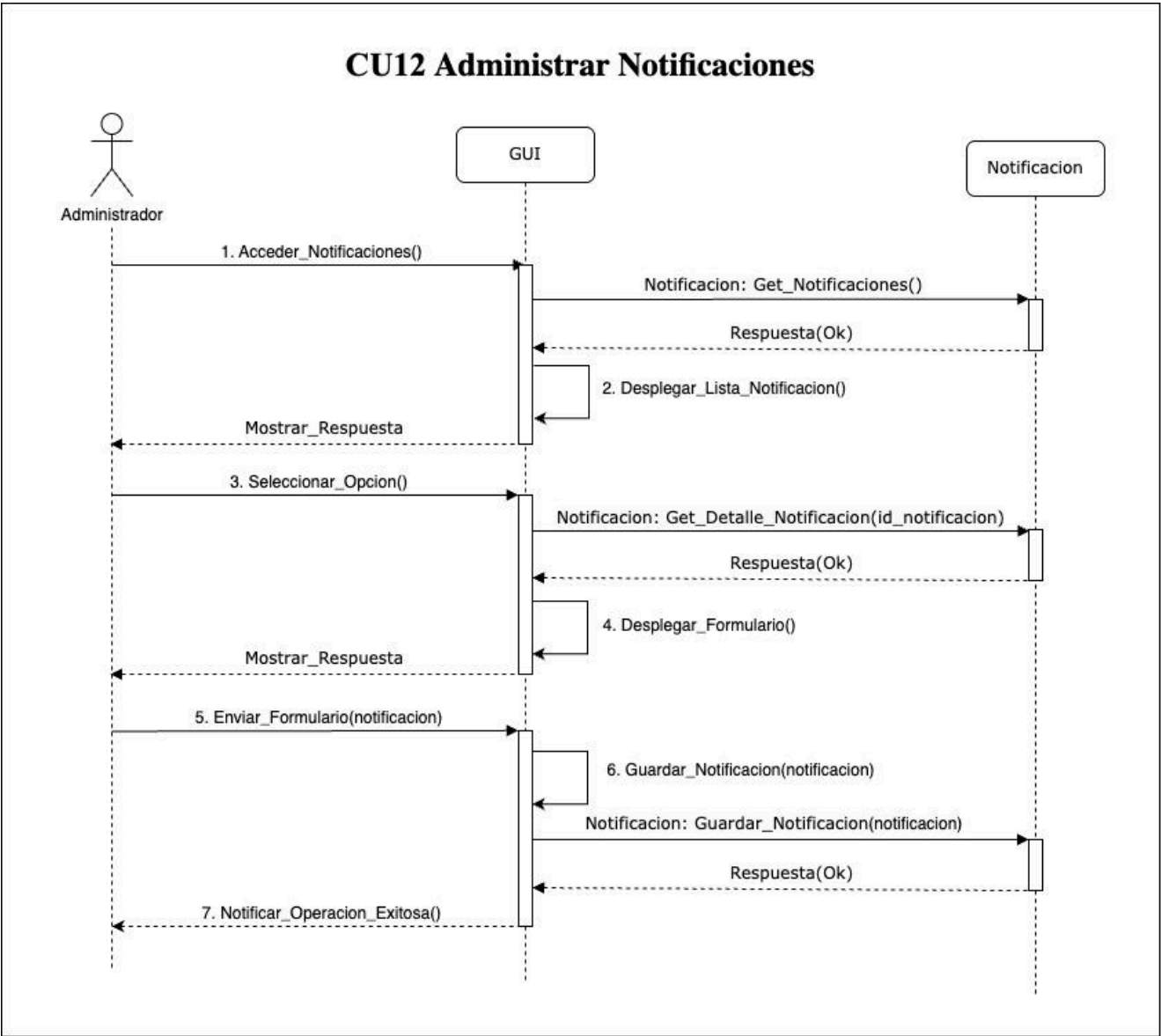


Figura 3.15. Diagrama de secuencia extendido: CU12 – Administrar Notificaciones
Fuente: Elaboración propia

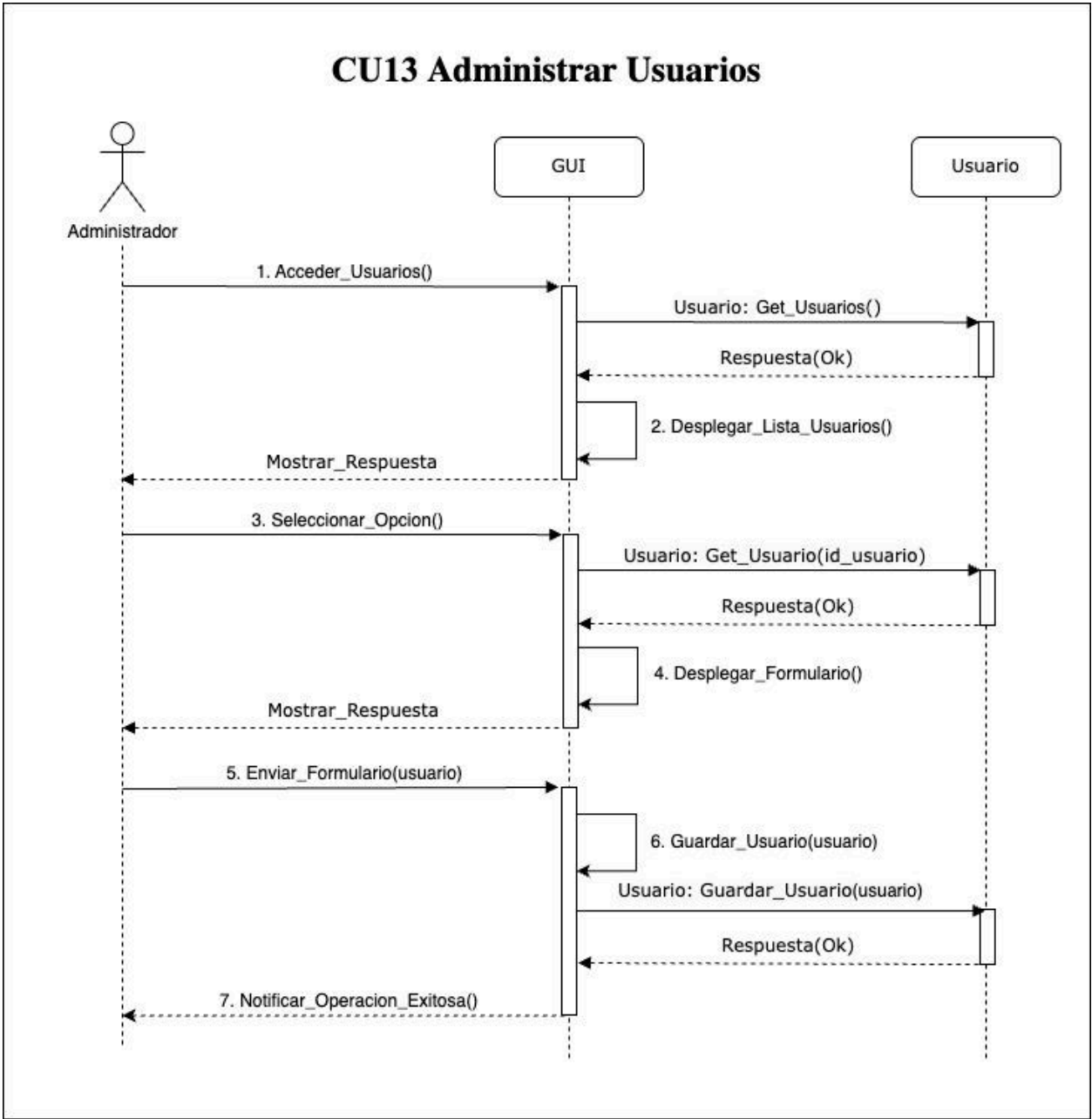


Figura 3.16. Diagrama de secuencia extendido: CU13 – Administrar Usuarios
Fuente: Elaboración propia

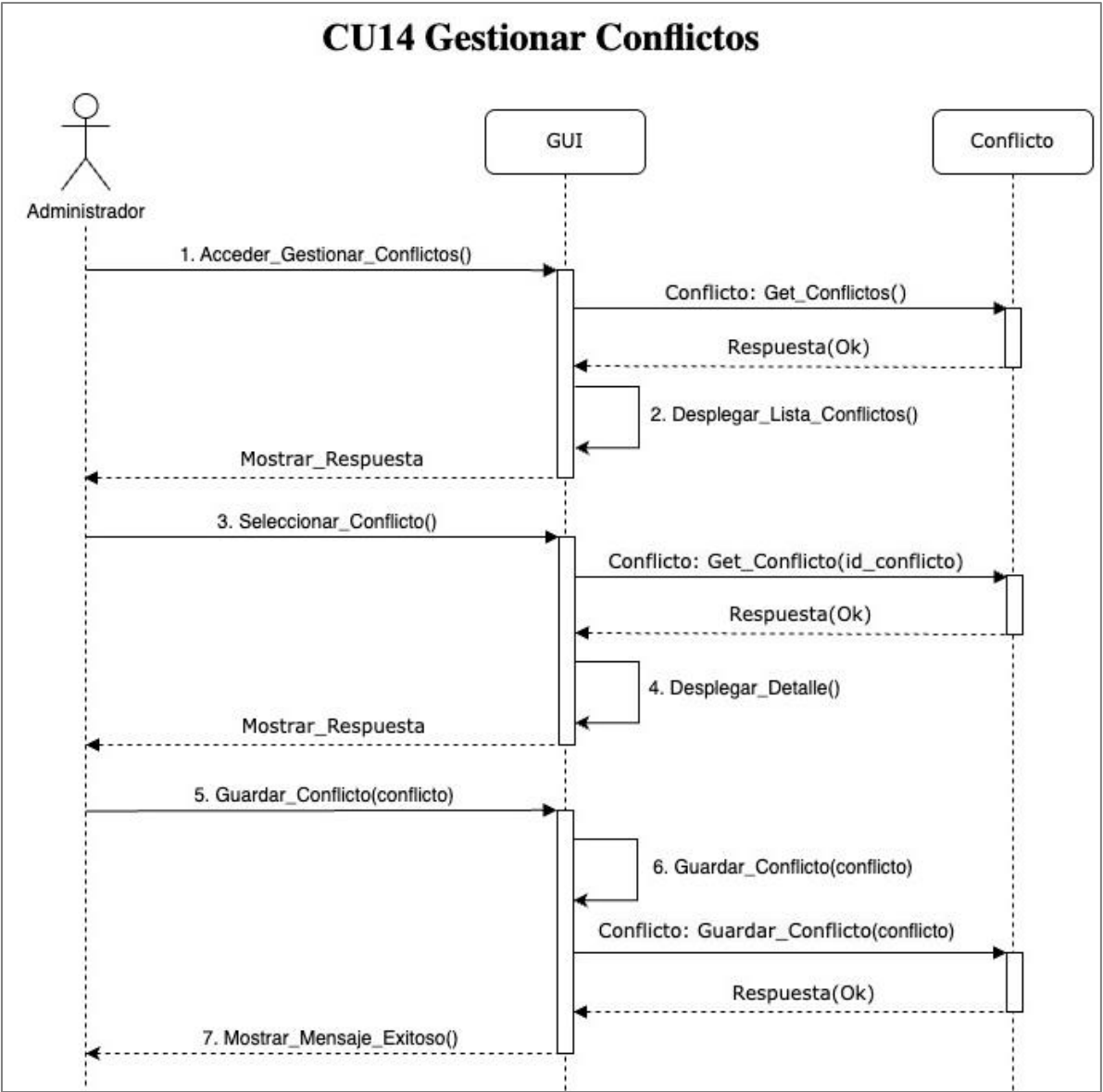


Figura 3.17. Diagrama de secuencia extendido: CU14 – Gestionar Conflictos
Fuente: Elaboración propia

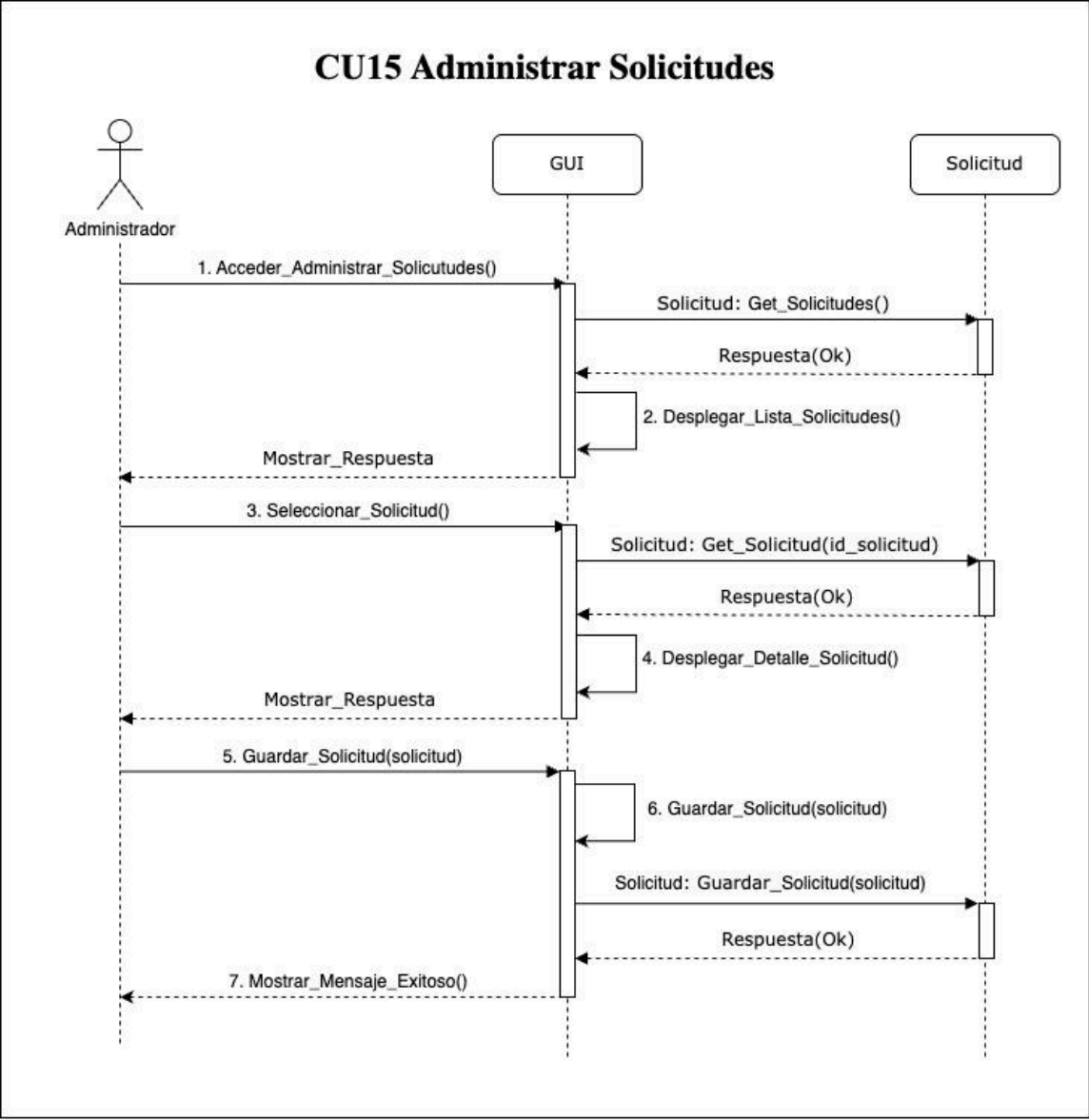


Figura 3.18. Diagrama de secuencia extendido: CU15 – Administrar Solicitudes
Fuente: Elaboración propia

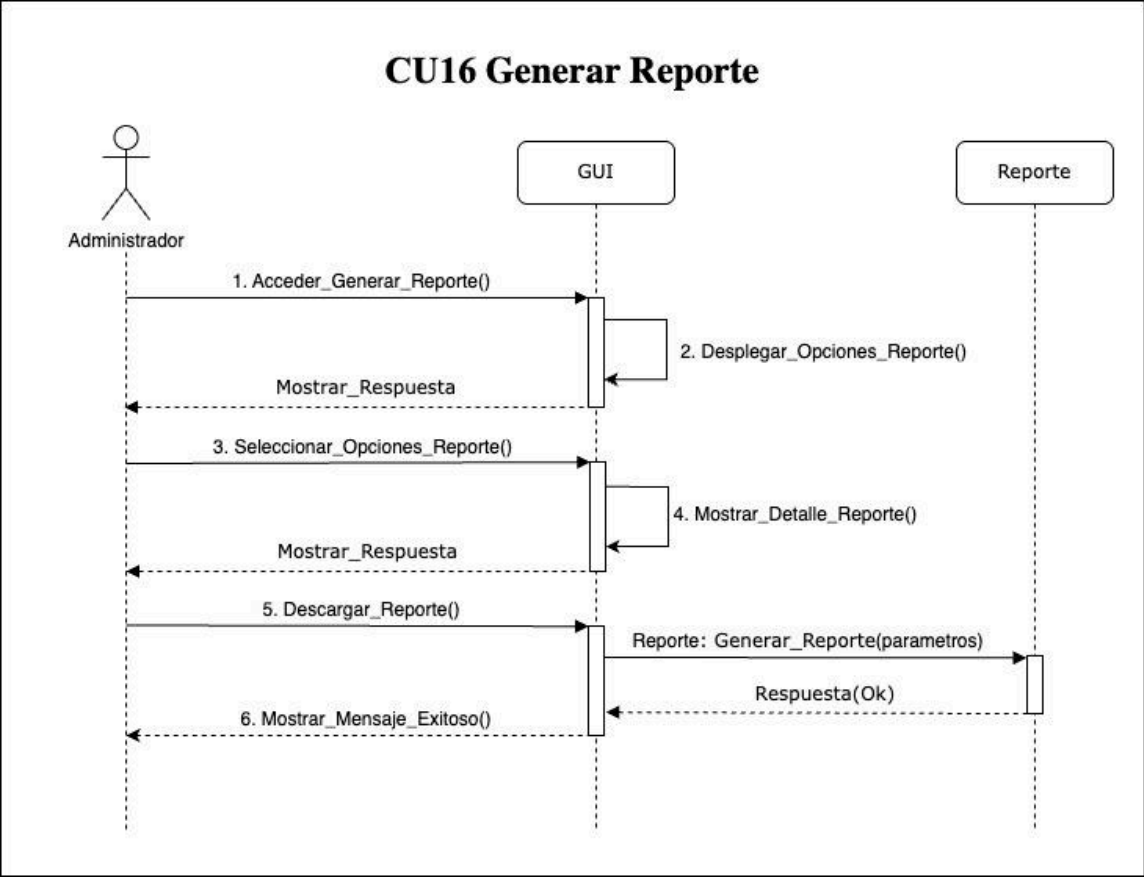


Figura 3.19. Diagrama de secuencia extendido: CU16 – Generar Reporte
Fuente: Elaboración propia

3.4. Diseño de interfaz

En la siguiente sección se indicarán algunas de las interfaces más importantes del sistema.

3.4.1. Inicio cliente

La pantalla de la Figura 3.20, corresponde al inicio que tendrán los Clientes una vez se hayan autenticado. En esta pantalla se podrá navegar entre las distintas opciones además de entregar información relevante para el sistema.

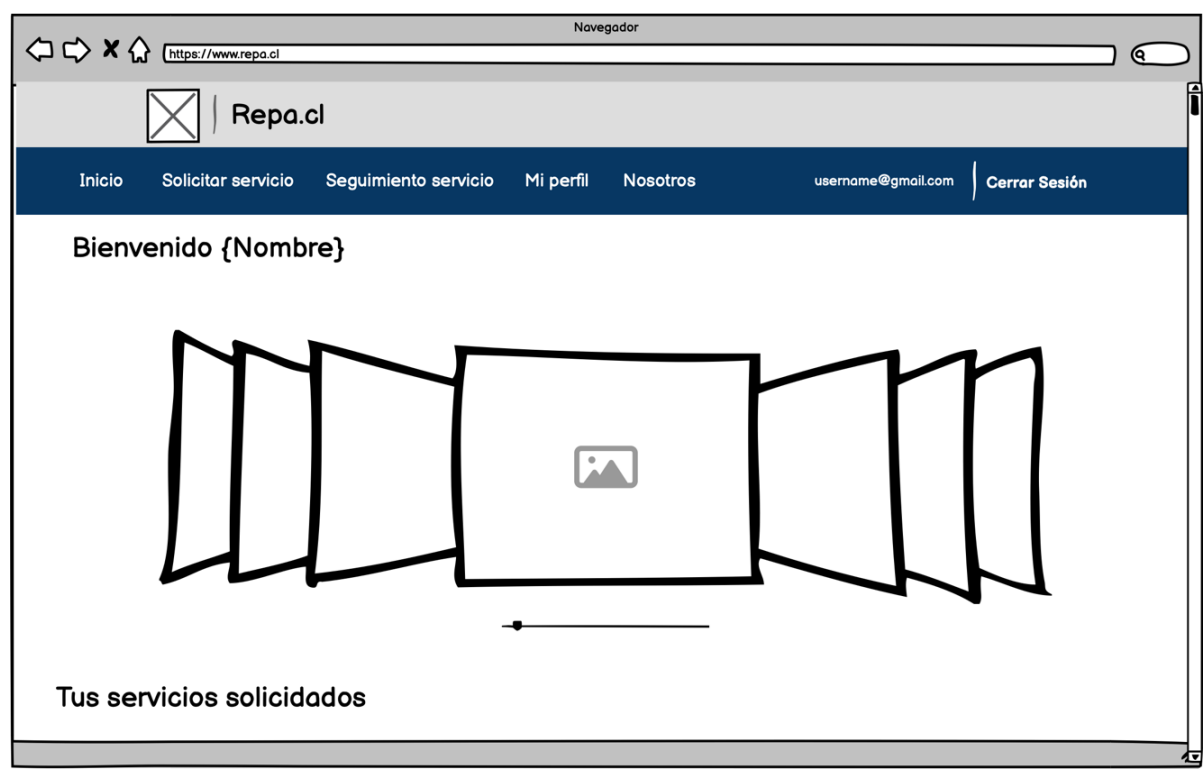


Figura 3.20. Interfaz Inicio Cliente
Fuente: Elaboración propia

Las Heurísticas de Usabilidad de Nielsen se usó para garantizar una experiencia intuitiva y eficiente. La visibilidad del estado del sistema se reflejó mostrando el nombre y correo electrónico del usuario autenticado en la parte superior. Se utilizó un lenguaje claro y familiar en los menús de navegación para facilitar la comprensión y alinearse con las expectativas de los usuarios. La barra de navegación siempre visible asegura el control y la libertad del usuario, mientras que el diseño minimalista prioriza únicamente los elementos esenciales, reduciendo la sobrecarga visual. Además, se consideró la prevención de errores al mostrar solo opciones relevantes tras la autenticación. Estas implementaciones permiten a los clientes interactuar con la plataforma de manera eficiente y segura.

3.4.2. Mi perfil – Cliente

La pantalla de la Figura 3.21, corresponde a los datos del Cliente registrado. Tendrán la facultad de cambiar su contraseña de manera sencilla y modificar los datos del Cliente.

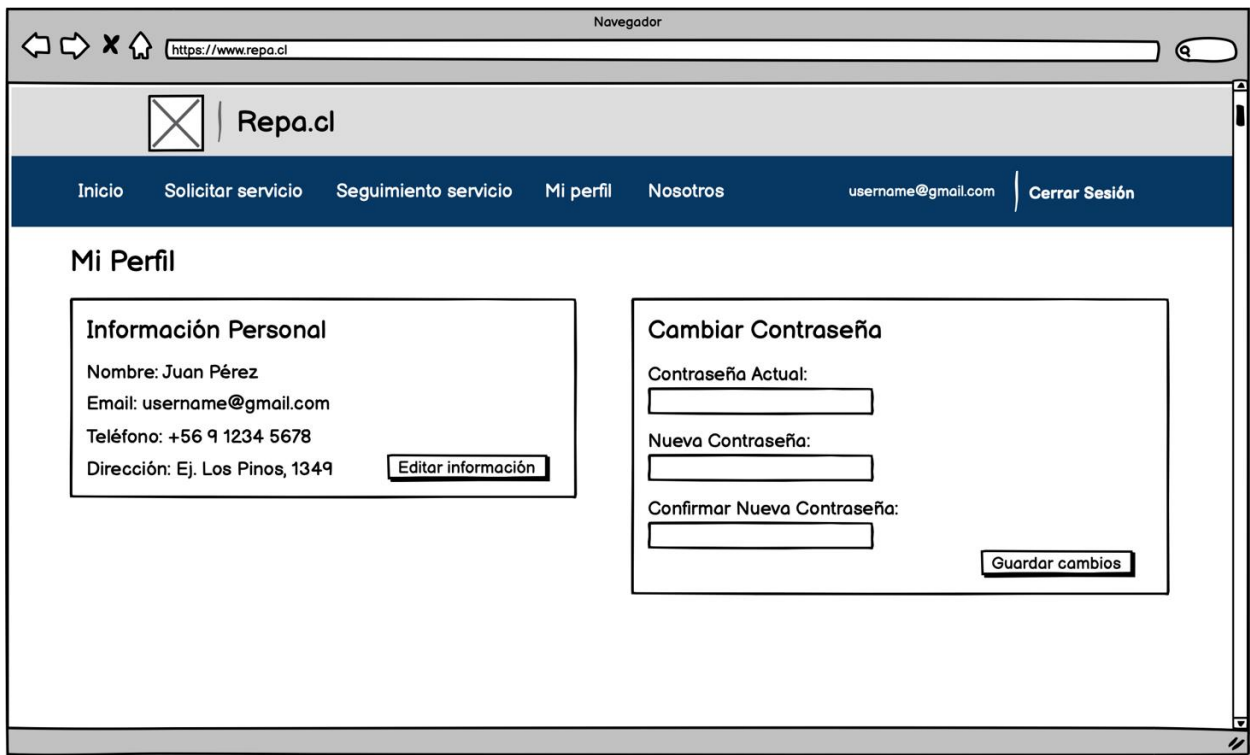


Figura 3.21. Interfaz Mi Perfil Cliente
Fuente: Elaboración propia

En el diseño de la interfaz "Mi Perfil" para los clientes, se implementó las Heurísticas de Usabilidad de Nielsen para optimizar la interacción del usuario con la plataforma. La visibilidad del estado del sistema se reflejó al mostrar los datos del cliente autenticado en un formato claro y accesible. Se utilizó un lenguaje sencillo y familiar en las etiquetas de los campos y botones, como "Editar información" y "Guardar cambios", alineándose con las expectativas de los usuarios. Además, la interfaz ofrece control y libertad al permitir al cliente actualizar tanto sus datos personales como su contraseña de manera independiente. El diseño minimalista organiza los elementos en dos secciones principales, lo que facilita la navegación y reduce la carga cognitiva. También se consideró la prevención de errores, como incluir campos separados para confirmar la nueva contraseña, minimizando posibles equivocaciones. Estas características aseguran que los usuarios puedan gestionar su perfil de manera eficiente y segura.

3.4.3. Ingresar solicitud servicio – Cliente

La pantalla de la figura 3.22, corresponde a la forma en que los clientes pueden crear sus solicitudes por medio de un formulario que tendrán que completar. Además, tiene un apartado de términos y condiciones en el que tendrán que aceptarlo antes de enviar la solicitud.

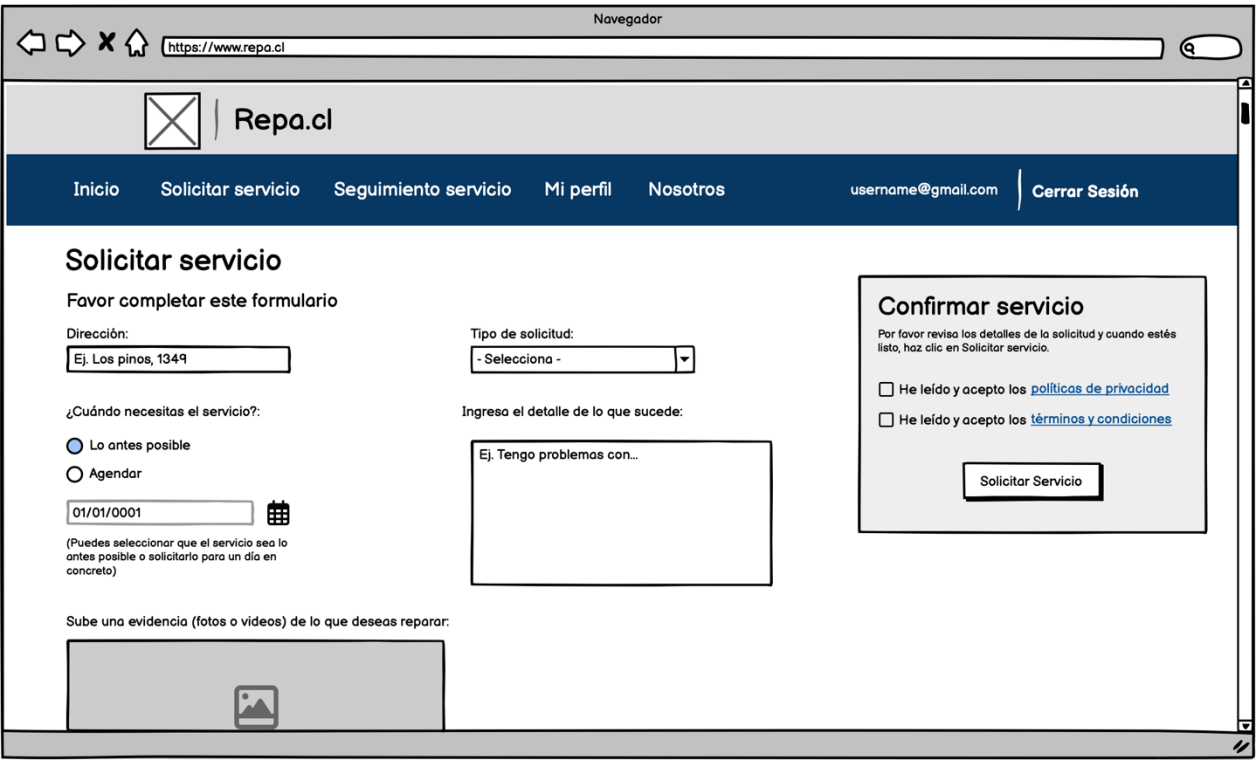


Figura 3.22. Interfaz Formulario Solicitud
Fuente: Elaboración propia

En el diseño de la interfaz para ingresar una solicitud de servicio, se aplicaron las Heurísticas de Usabilidad de Nielsen con el objetivo de asegurar una interacción clara y eficiente. La visibilidad del estado del sistema se representó mediante un formulario estructurado, en el que se destacaron las secciones requeridas para completar la solicitud. Se utilizó un lenguaje claro y familiar en los campos y botones, como "Solicitar Servicio" y "Sube una evidencia", asegurando que los términos fueran comprensibles para el usuario. La prevención de errores se implementó al exigir la aceptación de los términos y condiciones antes de permitir el envío de la solicitud. Además, se ofreció control al usuario al permitirle elegir entre agendar el servicio o solicitarlo de inmediato. El diseño se organizó de forma minimalista, reduciendo la sobrecarga cognitiva y presentando la información de manera clara. Estas implementaciones garantizaron una experiencia alineada con las necesidades de los clientes.

3.4.4. Ver solicitudes – Cliente

La pantalla de la figura 3.23, corresponde a la vista que tendrán los clientes en donde pueden ver sus solicitudes y el estado de ellas, además, si es que necesitan ver el detalle de la solicitud lo podrán revisar por medio del botón.

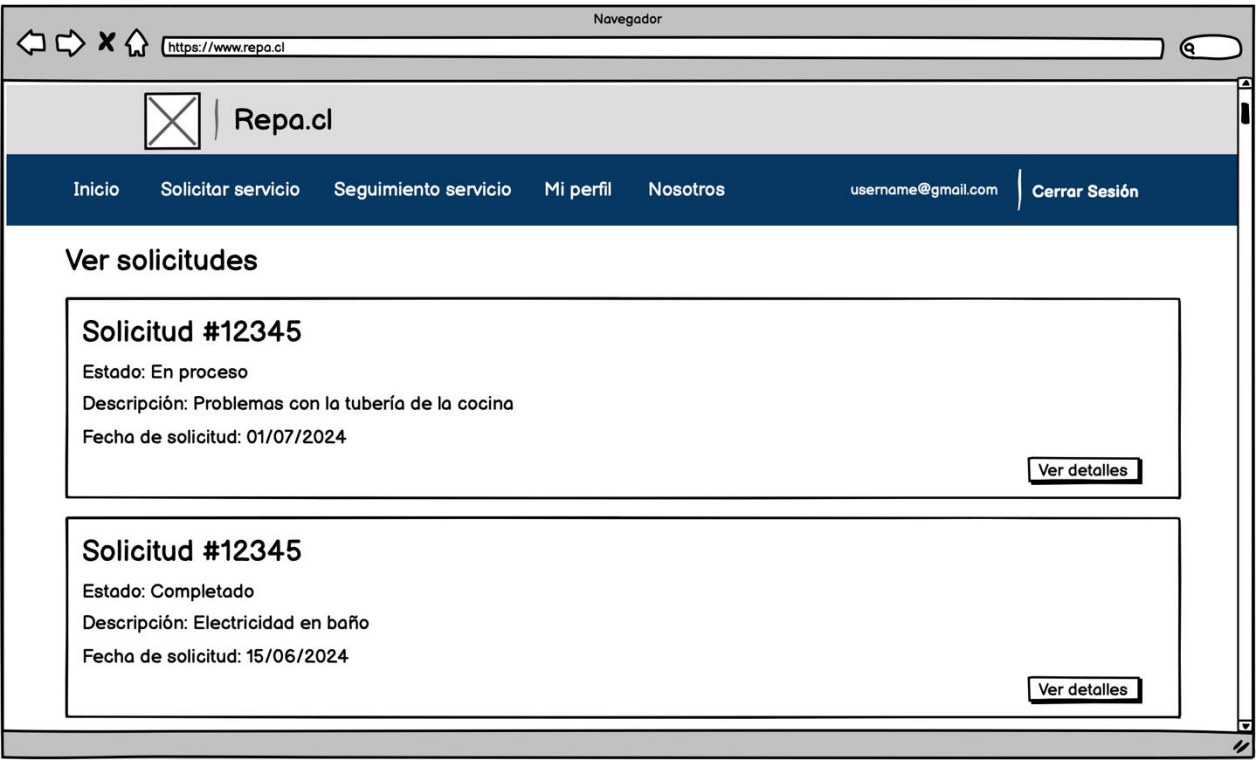


Figura 3.23. Interfaz Ver Solicitud Cliente
Fuente: Elaboración propia

En el diseño de la interfaz para ver solicitudes, se aplicaron las Heurísticas de Usabilidad de Nielsen con el propósito de facilitar la consulta y gestión de las solicitudes realizadas por los clientes. La visibilidad del estado del sistema se reflejó al mostrar claramente el estado de cada solicitud junto con información relevante como la descripción y la fecha de creación. Se utilizó un diseño minimalista para organizar las solicitudes en tarjetas individuales, lo que permitió al usuario identificar rápidamente la información de cada una. Además, se incluyó un botón etiquetado como "Ver detalles" para proporcionar control y libertad al usuario al acceder a información adicional cuando lo requiera. Estas características garantizaron una experiencia intuitiva y eficiente para la gestión de solicitudes dentro de la plataforma.

3.4.5. Ver solicitudes – Profesional

La pantalla de la figura 3.24, corresponde a la bandeja que tendrán los profesionales para recibir todas las solicitudes que suban los clientes y en ella podrán ver una vista previa de lo requerido. Además, constará con dos botones, el de ‘Ver Detalle’ y el de ‘Aceptar la solicitud’, esta última es esencial para continuar con el servicio.

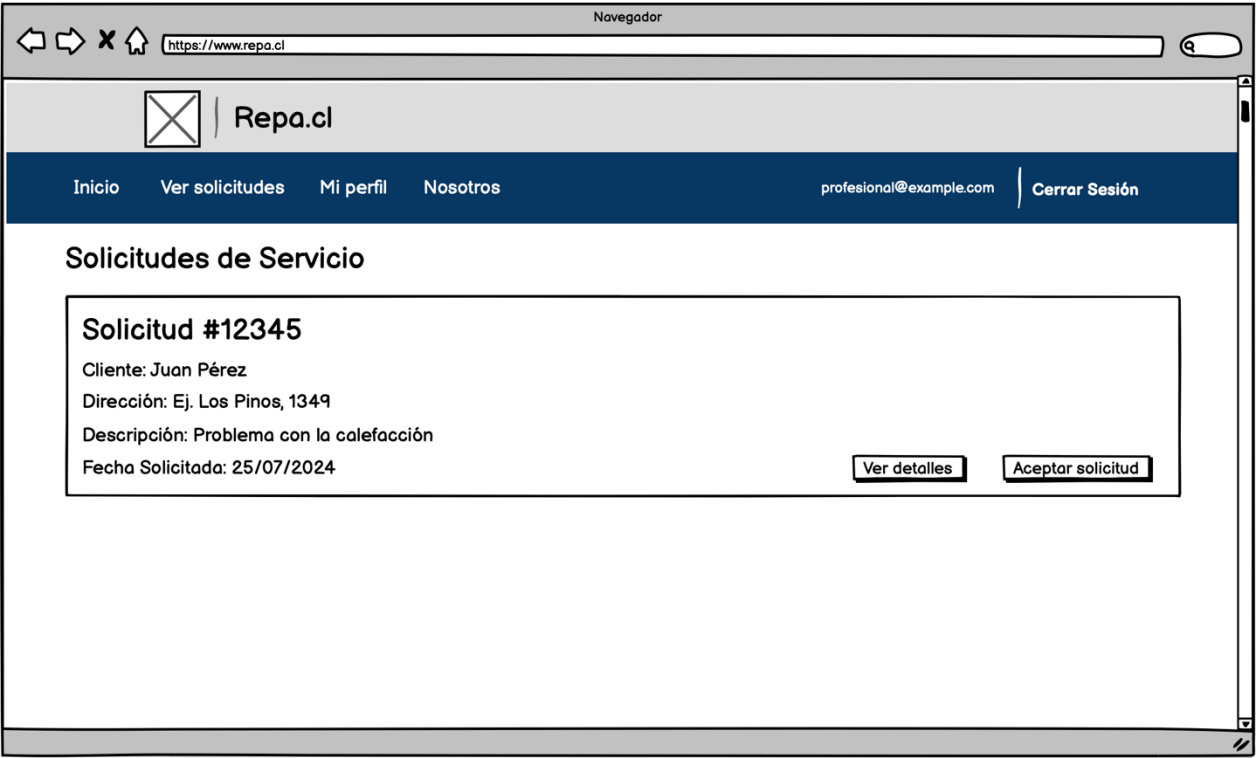


Figura 3.24. Ver Solicitudes Profesional
Fuente: Elaboración propia

En el diseño de la interfaz para ver solicitudes de los profesionales, se aplicaron las Heurísticas de Usabilidad de Nielsen con el fin de facilitar la gestión de las solicitudes enviadas por los clientes. La visibilidad del estado del sistema se aseguró mediante una vista previa clara de cada solicitud, que incluye información relevante como el cliente, la dirección, la descripción del problema y la fecha de solicitud. Se utilizó un diseño minimalista que organiza las solicitudes en tarjetas individuales para garantizar una rápida identificación y lectura de los datos. Además, se proporcionaron botones etiquetados como "Ver detalles" y "Aceptar solicitud", lo que otorga control y libertad al profesional para decidir cómo proceder con cada solicitud. Estas características garantizaron una experiencia eficiente para los profesionales al gestionar los servicios en la plataforma.

3.4.6. Ver detalle solicitud – Profesional

La pantalla de la figura 3.25, corresponde a la vista que tendrán los profesionales en donde podrán ver el detalle de las solicitudes que llegan, en ellas podrán ver los datos completos del usuario, la problemática y los archivos adjuntos.

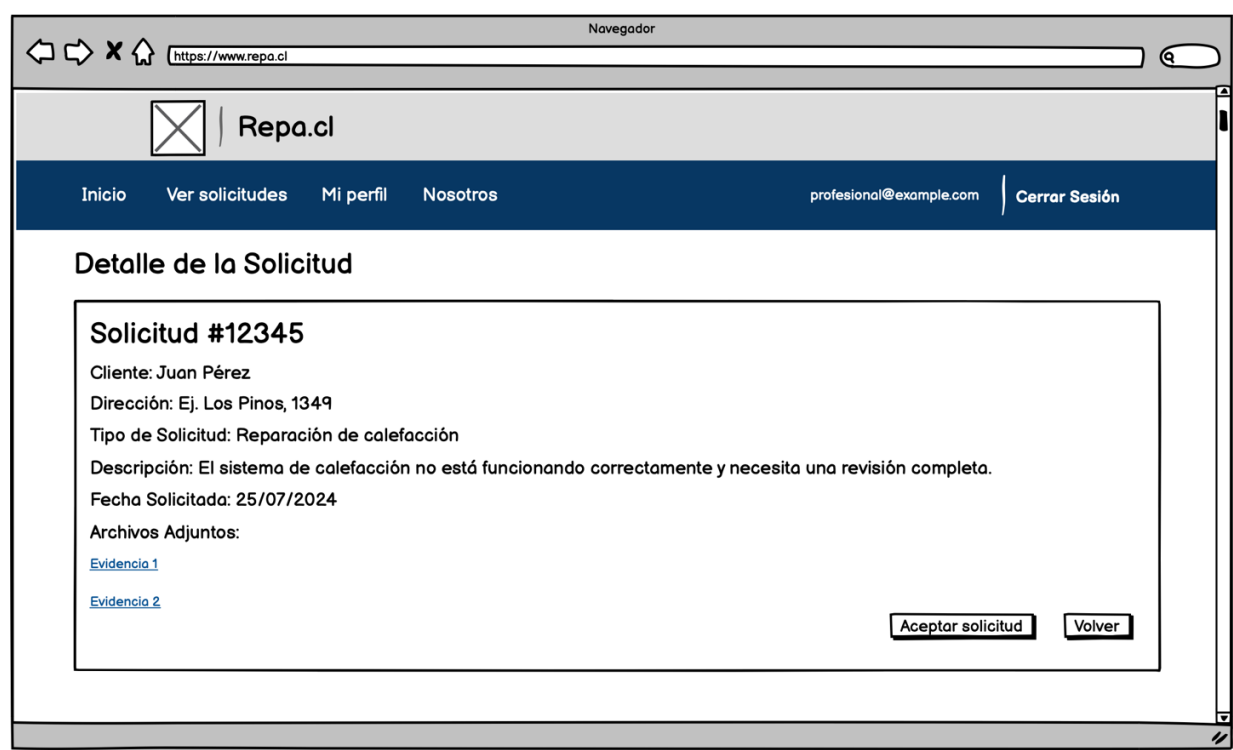


Figura 3.25. Interfaz Ver Detalle Solicitud Profesional
Fuente: Elaboración propia

En el diseño de la interfaz para ver el detalle de las solicitudes, se implementaron las Heurísticas de Usabilidad de Nielsen con el objetivo de proporcionar una experiencia clara y eficiente para los profesionales. La visibilidad del estado del sistema se aseguró mediante la presentación detallada de la información de cada solicitud, incluyendo los datos del cliente, la descripción del problema, la fecha de solicitud y los archivos adjuntos. Se utilizó un diseño organizado que permite al profesional identificar rápidamente la información relevante. El botón "Aceptar solicitud" brinda control al usuario para decidir cómo proceder con la solicitud, mientras que el botón "Volver" facilita la navegación dentro del sistema. Estas características garantizaron una experiencia intuitiva para la gestión de las solicitudes en detalle.

3.4.7. Inicio administrador

La pantalla de la figura 3.26, corresponde a la pantalla de inicio que tendrá el administrador, esta contendrá el acceso directo a los tres mantenedores que tiene el sistema (usuarios, servicios y estadísticas).

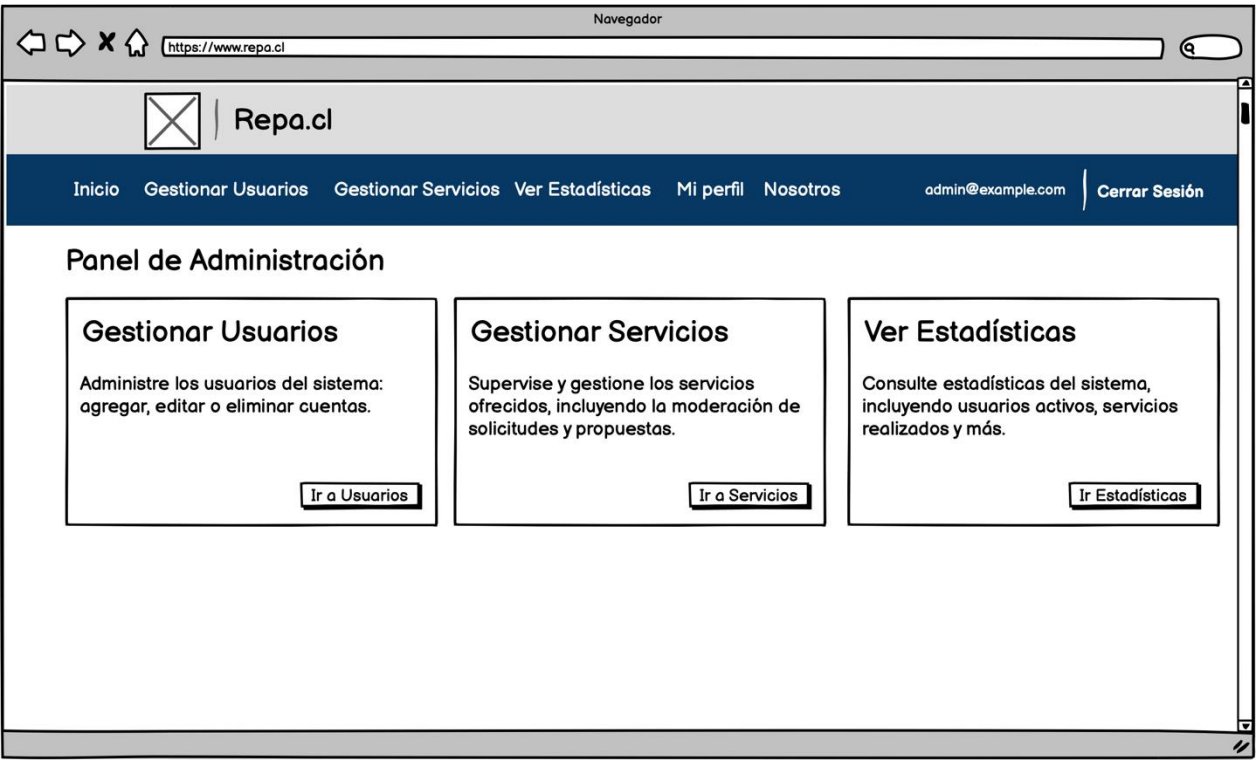


Figura 3.26. Interfaz Inicio Administrador
Fuente: Elaboración propia

En el diseño de la interfaz de inicio para el administrador, se aplicaron las Heurísticas de Usabilidad de Nielsen para garantizar una experiencia intuitiva y eficiente. La visibilidad del estado del sistema se implementó mediante un diseño claro que muestra las principales funcionalidades disponibles, organizadas en tarjetas individuales para "Gestionar Usuarios", "Gestionar Servicios" y "Ver Estadísticas". Cada tarjeta incluye una breve descripción de su función y un botón de acceso directo, proporcionando control y libertad al usuario para navegar hacia la sección deseada. Se utilizó un diseño minimalista que priorizó los elementos esenciales, reduciendo la carga cognitiva y facilitando la identificación rápida de las acciones. Estas características aseguraron una experiencia optimizada para el administrador en la gestión del sistema.

CONCLUSIONES

El desarrollo del proyecto de la Plataforma Web de Reparaciones Domiciliarias permite alcanzar los objetivos planteados, ofreciendo una solución que simplifica la búsqueda y contratación de profesionales para servicios de reparación y mantenimiento en el hogar. A través de una plataforma intuitiva y accesible, se garantiza que los usuarios logran gestionar solicitudes de manera eficiente y rápida, cumpliendo con el objetivo general del proyecto.

En relación con los objetivos específicos, se logra simplificar el acceso a profesionales calificados mediante un sistema que permitió evaluar sus calificaciones, tarifas y reseñas. Asimismo, se implementan funcionalidades que facilitan la gestión de reparaciones, desde la creación de solicitudes hasta la programación de visitas, asegurando transparencia y confianza en el proceso. La plataforma también proporciona a los usuarios estimaciones claras de costos antes de contratar un servicio, promoviendo una experiencia sin estrés.

Se garantiza que los profesionales registrados en la plataforma sean confiables, respaldados por un sistema de calificaciones y evaluaciones de otros usuarios. Además, se ofrecen opciones de comparación entre propuestas, fomentando decisiones informadas por parte de los usuarios. Finalmente, se reduce significativamente el tiempo y esfuerzo necesarios para encontrar y programar servicios, optimizando la experiencia de los usuarios y contribuyendo a mejorar la calidad de vida de los propietarios.

En conclusión, la plataforma no solo cumple con los objetivos establecidos, sino que también sentó las bases para un sistema escalable y adaptable a futuras mejoras. Se recomienda una continua exploración de nuevas funcionalidades y tecnologías que potencien la solución, consolidando el proyecto como una herramienta confiable y eficiente al momento de buscar reparaciones y mantenciones domiciliarias.

Sin embargo, aunque el proyecto busca promover la confianza entre propietarios y profesionales mediante herramientas tecnológicas, el éxito de esta solución depende, en cierta medida, en el compromiso ético y moral de sus usuarios. La honestidad, la responsabilidad y la transparencia son valores fundamentales que deben regir las interacciones dentro de la plataforma para garantizar que se mantenga la calidad del servicio y la satisfacción de los usuarios. Este llamado a la conciencia recuerda que, más allá de la tecnología, el respeto mutuo y la integridad son esenciales para construir un entorno de confianza y colaboración sostenible.

Desde una perspectiva formativa, este proyecto significó una oportunidad para desarrollar una visión integral sobre cómo se diseña, estructura y documenta un sistema de software. El proceso permitió comprender la importancia de traducir necesidades reales en soluciones

funcionales mediante el uso de metodologías, modelos y estándares técnicos. Asimismo, se reforzaron competencias clave como el análisis crítico, la toma de decisiones informadas y la capacidad de proponer soluciones escalables dentro de un marco realista y estructurado.

Esta experiencia también permitió valorar la importancia de una planificación bien fundamentada, de la trazabilidad entre objetivos y funcionalidades, y de la necesidad de diseñar soluciones centradas en las personas. Si bien no se llevó a cabo la implementación práctica del sistema, el trabajo aportó una base sólida para su futura construcción y representó una instancia significativa de crecimiento como profesional del área de la ingeniería.

BIBLIOGRAFÍA

- Sommerville, I. (2015). Software Engineering (10th ed.). Pearson.
- Fowler, M. (2002). Patterns of Enterprise Application Architecture. Addison-Wesley.
- Nielsen, J. (1994). Usability Engineering. Morgan Kaufmann.