**EstilosExpress**

**Informe Final**

**Portafolio De Título**

**Integrantes:** **Docente:**

Jorge Acuña Aguilera Mg. Jonathan López Acevedo

Mercideu Francois

Nicolas Ayarza Gatica

**Contenido**

[Resumen Ejecutivo 4](#_heading=h.gjdgxs)

[1.Propósito del Proyecto 4](#_heading=h.30j0zll)

[2.Objetivos 4](#_heading=h.1fob9te)

[3.Descripción de las Etapas del Proyecto 5](#_heading=h.3znysh7)

[4.Recursos Utilizados 5](#_heading=h.2et92p0)

[5.Resultados Esperados o Logrados 6](#_heading=h.tyjcwt)

[Introducción 7](#_heading=h.3dy6vkm)

[Descripción del Problema 9](#_heading=h.1t3h5sf)

[Solución al Problema 11](#_heading=h.4d34og8)

[1. Arquitectura del Proyecto 11](#_heading=h.2s8eyo1)

[2. Funcionalidades Clave 11](#_heading=h.17dp8vu)

[2.1 Sistema de Registro y Autenticación 11](#_heading=h.3rdcrjn)

[Objetivo General 15](#_heading=h.26in1rg)

[Objetivos Específicos 16](#_heading=h.lnxbz9)

[Competencias del Perfil de Egreso 19](#_heading=h.35nkun2)

[Asignación de Roles 21](#_heading=h.1ksv4uv)

[Metodología Utilizada en el Proyecto 25](#_heading=h.44sinio)

[Carta Gantt 29](#_heading=h.2jxsxqh)

[Implementación del Proyecto 29](#_heading=h.z337ya)

[ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE (ERS) 30](#_heading=h.3j2qqm3)

[1. Introducción 30](#_heading=h.1y810tw)

[1.1 Propósito 30](#_heading=h.4i7ojhp)

[1.2 Ámbito del Sistema 31](#_heading=h.2xcytpi)

[1.3 Acrónimos y Abreviaturas 31](#_heading=h.1ci93xb)

[1.4 Visión General del Documento 31](#_heading=h.3whwml4)

[2. Descripción General 32](#_heading=h.2bn6wsx)

[2.1 Perspectiva del Producto 32](#_heading=h.qsh70q)

[2.2 Funciones del Producto 32](#_heading=h.3as4poj)

[2.3 Características de los Usuarios 33](#_heading=h.1pxezwc)

[2.4 Restricciones 33](#_heading=h.49x2ik5)

[2.5 Suposiciones y Dependencias 33](#_heading=h.2p2csry)

[2.6 Requisitos Futuros 33](#_heading=h.147n2zr)

[3. Requisitos Específicos 34](#_heading=h.3o7alnk)

[3.1 Requisitos comunes de los interfaces 34](#_heading=h.23ckvvd)

[Interfaces de usuario 34](#_heading=h.ihv636)

[3.1.2 Interfaces de hardware 34](#_heading=h.32hioqz)

[3.1.3 Interfaces de software 34](#_heading=h.1hmsyys)

[3.1.4 Interfaces de comunicación 34](#_heading=h.41mghml)

[3.2 Requisitos Funcionales 34](#_heading=h.2grqrue)

[3.3 Requisitos No Funcionales Rendimiento: 43](#_heading=h.vx1227)

[3.4 Otros Requisitos 45](#_heading=h.3fwokq0)

[Modelo 4+1 45](#_heading=h.1v1yuxt)

[Requerimientos Técnicos 65](#_heading=h.4f1mdlm)

[Desarrollo de la Solución 69](#_heading=h.2u6wntf)

[1. Configuración de la Base de Datos 70](#_heading=h.19c6y18)

[2. Implementación del Backend 72](#_heading=h.3tbugp1)

[3. Configuración del Frontend 73](#_heading=h.28h4qwu)

[4. Integración de Servicios Externos 74](#_heading=h.nmf14n)

[5. Pruebas Técnicas 74](#_heading=h.37m2jsg)

[Factibilidad Económica 75](#_heading=h.1mrcu09)

[Conclusión (Español) 76](#_heading=h.46r0co2)

[Acrónimos 77](#_heading=h.2lwamvv)

[Bibliografía 77](#_heading=h.111kx3o)

# Resumen Ejecutivo

## 1.Propósito del Proyecto

El proyecto **EstilosExpress** surge como respuesta a dos tendencias crecientes en la sociedad actual. Por un lado, el aumento significativo en el porcentaje de familias que tienen mascotas, especialmente perros, ha generado una mayor demanda de servicios especializados como cortes de pelo, baños sanitarios y cortes de uñas. Según estudios recientes, más del 60% de los hogares poseen una mascota, y los dueños están cada vez más interesados en brindarles cuidados estéticos y de higiene regulares.

Por otro lado, es cada vez más común que hombres y mujeres busquen estilistas y barberos cercanos a sus domicilios, priorizando la comodidad y el acceso rápido a servicios de calidad. Sin embargo, la falta de una plataforma que centralice la búsqueda y reserva de estos servicios representa una barrera tanto para los usuarios como para los profesionales independientes que desean ofrecer sus servicios.

En este contexto, **EstilosExpress** se posiciona como una solución integral que conecta a los usuarios con estilistas, barberos y peluqueros caninos cercanos, ofreciendo una experiencia sencilla, eficiente y adaptada a las necesidades actuales.

## 2.Objetivos

El objetivo principal del proyecto es desarrollar una plataforma que facilite a los usuarios la búsqueda y reserva de servicios de peluquería personalizados, garantizando rapidez y eficiencia en el proceso. Los objetivos específicos incluyen el desarrollo de un sistema de registro para usuarios y profesionales, la implementación de una función de geolocalización que permita identificar a profesionales cercanos y el diseño de un sistema de reservas que gestione horarios de manera organizada y efectiva.

## 3.Descripción de las Etapas del Proyecto

El desarrollo del proyecto se llevó a cabo en las siguientes etapas:

* Planificación: identificación de requisitos y creación de Mockups.
* Diseño: desarrollo de prototipos y estructura de la interfaz.
* Desarrollo: implementación del sistema de registro, geolocalización y reservas.
* Pruebas: validación de funcionalidades con usuarios potenciales.
* Entrega: presentación final del MVP

## 4.Recursos Utilizados

El proyecto fue llevado a cabo por un equipo conformado por tres integrantes, quienes colaboraron estrechamente para cumplir con los objetivos establecidos. Cada miembro del equipo asumió roles específicos y complementarios, lo que permitió abordar las diferentes áreas del desarrollo con eficacia y organización. Para garantizar un desarrollo moderno y funcional, se utilizaron tecnologías avanzadas como Angular, que sirvió como el marco principal para la construcción de la lógica de la aplicación, e Ionic, que facilitó la creación de una interfaz amigable y adaptable a dispositivos móviles. Asimismo, Supabase se utilizó para la gestión eficiente de la base de datos y Leaflet se integró como herramienta clave para la funcionalidad de geolocalización.

En cuanto a los recursos financieros, estos fueron gestionados de manera interna, destacándose la importancia de optimizar el presupuesto disponible mediante una cuidadosa planificación. Se elaboró un flujo de caja que incluyó los gastos necesarios, pero en proporciones bajas, priorizando el uso eficiente de los recursos. Para maximizar la viabilidad económica del proyecto, se optó por herramientas de código abierto, lo que permitió acceder a soluciones tecnológicas de alta calidad sin incurrir en costos significativos. Esta estrategia no solo favoreció un control riguroso de los gastos, sino que también aseguró que los recursos financieros se destinaran a las áreas más críticas del desarrollo. Como resultado, el equipo logró construir una plataforma robusta y escalable que cumpliera con los requerimientos establecidos, respetando los límites presupuestarios definidos.

## 5.Resultados Esperados o Logrados

El resultado esperado del proyecto es una aplicación móvil diseñada para simplificar significativamente el proceso de búsqueda y reserva de servicios de peluquería personalizados. Esta plataforma está pensada para optimizar la experiencia del usuario, brindándole herramientas intuitivas y accesibles que permitan localizar rápidamente estilistas, barberos y peluqueros caninos cercanos, así como agendar citas de manera eficiente. Al mismo tiempo, el proyecto busca beneficiar a los profesionales independientes, proporcionándoles una solución integral que no solo facilite la gestión de sus citas, sino que también les permita aumentar su visibilidad y captar nuevos clientes de forma efectiva.

La aplicación no solo pretende conectar a los usuarios con los profesionales adecuados, sino también ofrecer una experiencia adaptada a las necesidades específicas de ambos grupos. Por un lado, los usuarios podrán acceder a un sistema organizado que incluye geolocalización, información detallada de los servicios ofrecidos y disponibilidad en tiempo real. Por otro lado, los profesionales tendrán acceso a herramientas que les permitirán administrar su agenda, optimizar su tiempo y mejorar la calidad del servicio ofrecido.

Este enfoque integral garantiza que la plataforma no solo sea funcional, sino también escalable, permitiendo su crecimiento y adaptación a las demandas del mercado. Con ello, EstilosExpress busca posicionarse como una solución innovadora que revolucione la forma en que se gestionan los servicios de peluquería, ofreciendo una experiencia eficiente, moderna y personalizada tanto para usuarios como para profesionales.

# Introducción

En los últimos años, hemos observado un crecimiento significativo en la tenencia de mascotas, especialmente perros, así como un aumento en la cantidad de personas que frecuentan barberías y salones de belleza. Estos cambios reflejan una mayor demanda de servicios personalizados, tanto para el cuidado de mascotas como para el ámbito estético. Sin embargo, hemos identificado diversas problemáticas asociadas a la oferta y disponibilidad de estos servicios. Por ejemplo, es común que los usuarios deban esperar hasta dos horas para recibir atención en barberías o salones debido a la alta demanda, lo que afecta negativamente su experiencia. Al mismo tiempo, existe una gran cantidad de profesionales que no cuentan con un local establecido ni con la infraestructura necesaria para abrir uno, limitando sus oportunidades laborales. Adicionalmente, muchas personas prefieren por comodidad que estos servicios sean prestados a domicilio, una opción que no siempre está disponible.

El proyecto **EstilosExpress** nace con el propósito de abordar estas necesidades mediante el desarrollo de una aplicación móvil que conecte a usuarios con estilistas, barberos y peluqueros caninos independientes. Los principales usuarios de esta plataforma serán dueños de mascotas, especialmente perros, quienes necesitan servicios como cortes de pelo, baños sanitarios y cortes de uñas. También está dirigida a estilistas que están comenzando en el mundo profesional, barberos que buscan independizarse y profesionales que necesitan manejar sus tiempos de trabajo de forma flexible debido a estudios u otros compromisos personales.

En esta etapa, el objetivo es presentar un MVP (Producto Mínimo Viable) funcional que permita demostrar las capacidades principales de la aplicación para la fecha de evaluación. El desarrollo completo del proyecto se proyecta para junio de 2025, pero este avance inicial se enfocará en funcionalidades clave como registro de usuarios, geolocalización, búsqueda de profesionales y un sistema de reservas.

Para lograrlo, hemos utilizado tecnologías modernas como Angular, Ionic, TypeScript, Leaflet y Supabase. Estas herramientas son ideales para trabajar en aplicaciones móviles debido a su capacidad para construir interfaces intuitivas, manejar datos en tiempo real y garantizar una experiencia de usuario fluida y escalable. Además, estas tecnologías permiten una implementación ágil y adaptable, alineada con las necesidades del mercado actual.

Con EstilosExpress, buscamos generar cambios significativos en la forma en que se gestionan estos servicios. Los usuarios tendrán una experiencia más cómoda al poder buscar y contratar servicios desde la comodidad de su hogar, con acceso a múltiples opciones que se adapten a sus necesidades. Por otro lado, los profesionales independientes tendrán una herramienta que les permita visibilizar su trabajo, gestionar sus horarios y construir su cartera de clientes de manera flexible y efectiva. En un mundo donde prácticamente todas las personas tienen acceso a un teléfono móvil, este proyecto busca aprovechar esta conectividad para facilitar la vida tanto de los usuarios como de los profesionales.

# Descripción del Problema

En los últimos años, se ha observado un cambio en las dinámicas sociales y económicas que ha incrementado la demanda de servicios de peluquería tanto para personas como para mascotas. Por un lado, el crecimiento de hogares con mascotas, especialmente perros, ha generado una mayor necesidad de servicios especializados como cortes de pelo, baños sanitarios y mantenimiento general. **Según un informe del Instituto Nacional de Estadísticas (INE) en 2022, el 74% de los hogares en Chile que tienen mascotas poseen perros**, lo que refleja una tendencia creciente hacia la tenencia responsable de animales. Sin embargo, muchos dueños enfrentan dificultades para encontrar profesionales que ofrezcan estos servicios, especialmente a domicilio.

Por otro lado, el mercado de peluquería y barbería para personas también ha experimentado un auge. **Según datos del Global Wellness Institute, este sector ha crecido un 10% anual desde 2020**, impulsado por una mayor demanda de servicios personalizados. A pesar de este crecimiento, las barberías y salones de belleza suelen estar saturados, lo que obliga a los usuarios a esperar hasta dos horas para recibir atención. Esta situación no solo afecta la experiencia del cliente, sino que también limita su capacidad para acceder a servicios cercanos y convenientes.

Además, existe una problemática importante para los profesionales del sector. Muchos estilistas y barberos independientes no cuentan con los recursos necesarios para abrir un local propio, lo que reduce su visibilidad y su capacidad para captar clientes. Por otro lado, aquellos que recién comienzan en el mercado profesional enfrentan desafíos para gestionar sus horarios de manera eficiente o para promocionar sus servicios en un entorno competitivo. **Según una encuesta realizada por Think With Google**, el 70% de los usuarios prefieren contratar servicios cercanos que puedan ser ofrecidos de manera flexible, incluyendo opciones a domicilio.

En este contexto, surge la necesidad de una solución que facilite la conexión entre usuarios y profesionales independientes, abordando las siguientes problemáticas:

1. Los usuarios necesitan opciones más cómodas y personalizadas, que les permitan acceder a servicios de peluquería sin largos tiempos de espera y, en muchos casos, desde la comodidad de su hogar.
2. Los profesionales requieren una plataforma que les permita visibilizar su trabajo, captar nuevos clientes y gestionar sus horarios de manera efectiva, especialmente aquellos que trabajan sin un local establecido.

Estas problemáticas, amplificadas por la creciente conectividad y el acceso masivo a teléfonos móviles, hacen evidente la necesidad de un proyecto como EstilosExpress. La falta de una herramienta integral que centralice la búsqueda, reserva y gestión de servicios de peluquería afecta tanto a los usuarios finales como a los profesionales, limitando las oportunidades de crecimiento en un mercado con alta demanda.

# Solución al Problema

El proyecto EstilosExpress propone una solución integral basada en el desarrollo de una aplicación móvil que conecta a usuarios con estilistas, barberos y peluqueros caninos independientes. Esta plataforma está diseñada con un enfoque técnico que prioriza la eficiencia, la escalabilidad y la experiencia del usuario, resolviendo las problemáticas actuales relacionadas con la búsqueda, la reserva y la gestión de servicios de peluquería.

### 1. Arquitectura del Proyecto

La arquitectura MVC es un patrón de diseño que separa la aplicación en tres componentes principales: Modelo, Vista y Controlador, cada uno con una responsabilidad específica:

* **Modelo:** En nuestro caso, el Modelo está representado principalmente por Supabase, que gestiona las bases de datos y los datos de la aplicación, tales como los usuarios registrados, los profesionales, las reservas y la geolocalización. Supabase actúa como un servicio backend, manejando la lógica de acceso a datos y la autenticación de usuarios.
* **Vista:** La Vista está compuesta por los componentes de la interfaz de usuario en la aplicación web y móvil, desarrollados utilizando Angular e Ionic. La Vista es la capa que se encarga de mostrar los datos a los usuarios y de interactuar con ellos a través de formularios, botones y otros elementos visuales. Esta capa también presenta el mapa de geolocalización donde los usuarios pueden buscar profesionales cercanos.
* **Controlador:** El Controlador está representado por los componentes y servicios de Angular. Estos manejan la lógica de negocio de la aplicación, como el procesamiento de datos del usuario, la interacción con los servicios de geolocalización y la gestión de las reservas. Los controladores reciben las entradas del usuario desde la vista y se comunican con el modelo para obtener o actualizar los datos según sea necesario.

Este enfoque MVC garantiza una estructura limpia y modular, lo que facilita el mantenimiento, la escalabilidad y la reutilización de código, al mismo tiempo que asegura una separación clara de responsabilidades.

### 2. Funcionalidades Clave

### 2.1 Sistema de Registro y Autenticación

El sistema de registro permite que tanto los usuarios como los profesionales creen perfiles personalizados. La autenticación está gestionada por Supabase Auth, que utiliza estándares de seguridad como OAuth 2.0 y JWT (JSON Web Tokens). Este enfoque garantiza:

* Un acceso seguro a las cuentas.
* La validación de datos en tiempo real, evitando errores o duplicaciones.
* La capacidad de recuperar contraseñas de manera sencilla.

Además, los perfiles incluyen información relevante, como ubicación, servicios ofrecidos y disponibilidad, que se almacenan en la base de datos relacional gestionada por Supabase.

**2.2 Geolocalización y Búsqueda de Profesionales Cercanos**

La funcionalidad de geolocalización se implementa con Leaflet, una biblioteca de mapas interactivos, integrada con las coordenadas GPS del dispositivo móvil. Este módulo realiza las siguientes operaciones técnicas:

* Obtención precisa de la ubicación del usuario mediante la API de geolocalización del dispositivo.
* Consulta optimizada a la base de datos para identificar a los profesionales dentro de un radio predefinido.
* Representación visual de los resultados en un mapa interactivo, con marcadores personalizados para cada profesional.
* Además, se ofrece la posibilidad de calcular rutas hacia el profesional seleccionado utilizando enlaces a plataformas como Google Maps o Waze, mejorando la experiencia del usuario.

**2.3 Sistema de Reservas con Gestión de Horarios**

El sistema de reservas se basa en un modelo de bloques de tiempo, donde cada profesional define su disponibilidad semanal. Este módulo incluye:

* Validación automática de la disponibilidad en tiempo real.
* Gestión de conflictos para evitar que dos usuarios reserven el mismo horario.
* Notificaciones automáticas para confirmar las reservas o informar sobre cambios.
* El diseño del sistema asegura que los usuarios puedan realizar reservas de manera rápida, mientras los profesionales tienen control total sobre su agenda.

**3. Tecnologías Utilizadas**

**3.1 Angular e Ionic**

Angular se utiliza como framework principal para el desarrollo del frontend, asegurando una lógica de aplicación robusta y organizada.

Ionic, junto con Capacitor, permite desarrollar una experiencia nativa en dispositivos móviles, con interfaces responsivas y optimizadas para diferentes tamaños de pantalla.

**3.2 Supabase**

Supabase actúa como el backend del proyecto, ofreciendo un sistema de gestión de datos basado en PostgreSQL. Esta solución permite realizar consultas rápidas y escalables, garantizando que la aplicación pueda manejar un gran volumen de usuarios.

**3.3 Leaflet**

Leaflet proporciona una integración eficiente de mapas, permitiendo la representación interactiva de datos espaciales con bajo impacto en el rendimiento.

**3.4 TypeScript**

TypeScript asegura un desarrollo más seguro y mantenible, reduciendo los errores comunes y mejorando la calidad del código.

**4. Procesos de Validación y Pruebas**

**4.1 Pruebas Unitarias y de Integración**

Cada módulo funcional se somete a pruebas unitarias para validar que se comporta según lo esperado. Posteriormente, se realizan pruebas de integración para verificar que los módulos interactúan correctamente entre sí.

**4.2 Pruebas de Rendimiento**

Se implementan técnicas de optimización para garantizar tiempos de carga mínimos, como la carga diferida de componentes y la optimización de consultas a la base de datos.

**4.3 Pruebas de Seguridad**

Los datos sensibles, como contraseñas y tokens, se cifran antes de almacenarse.

Se realizan auditorías de seguridad para identificar y solucionar vulnerabilidades potenciales.

**5. Gestión de Recursos y Viabilidad**

El desarrollo del proyecto se lleva a cabo por un equipo de tres integrantes con roles definidos: diseño de interfaz, desarrollo técnico y gestión del proyecto. Los recursos financieros son gestionados internamente, con un enfoque en la utilización de herramientas de código abierto para optimizar costos sin comprometer la calidad.

Además, la modularidad del diseño garantiza que nuevas funcionalidades puedan ser agregadas en el futuro, como pagos en línea, evaluaciones de los servicios y una expansión hacia la web. Este enfoque asegura que el proyecto sea viable no solo en su estado actual, sino también en su evolución futura.

**6. Impacto Esperado**

Con la implementación de EstilosExpress, los usuarios podrán acceder a una solución integral que les permita buscar, comparar y reservar servicios de peluquería personalizados de manera sencilla y eficiente. Los profesionales independientes, por su parte, obtendrán una herramienta poderosa para gestionar su tiempo, promocionar sus servicios y expandir su base de clientes.

Esta solución técnica no solo resuelve las problemáticas actuales, sino que también establece un estándar moderno para la digitalización de los servicios de peluquería, aprovechando la conectividad móvil para generar un impacto positivo en ambos lados del mercado.

# Objetivo General

Diseñar e implementar una aplicación móvil integral que permita a los usuarios buscar, localizar y reservar servicios de peluquería personalizados de manera eficiente, resolviendo las problemáticas actuales de accesibilidad, conveniencia y visibilidad de los profesionales independientes. La solución aprovechará tecnologías modernas como Angular, Ionic y Supabase para garantizar una experiencia fluida y escalable, con una arquitectura modular que facilite la integración de funcionalidades avanzadas.

Entre estas funcionalidades se incluye un sistema de autenticación y registro de usuarios seguro, mapas interactivos con Leaflet para la geolocalización en tiempo real de profesionales cercanos, y un sistema de reservas dinámico que gestione horarios y disponibilidad de manera organizada. Además, la plataforma estará optimizada para dispositivos móviles, ofreciendo tiempos de respuesta rápidos y una interfaz adaptativa que se ajuste a diferentes tamaños de pantalla y recursos del dispositivo.

El proyecto tiene como propósito principal mejorar significativamente la experiencia del usuario, brindándole la posibilidad de acceder a servicios de peluquería personalizados desde la comodidad de su hogar o en ubicaciones cercanas, con múltiples opciones para elegir según sus necesidades. Por otro lado, busca empoderar a los profesionales independientes, proporcionándoles herramientas efectivas para gestionar sus agendas, aumentar su visibilidad en el mercado y captar clientes de manera más eficiente, especialmente aquellos que no cuentan con un local establecido.

Además de resolver las necesidades actuales, el desarrollo de esta solución sentará las bases para la expansión futura del proyecto, permitiendo la incorporación de nuevas funcionalidades como pagos en línea y evaluaciones de usuarios. Se adoptarán prácticas de desarrollo sostenibles que aseguren la viabilidad técnica y económica a largo plazo, garantizando que la plataforma evolucione junto con las demandas del mercado y las expectativas de los usuarios.

# Objetivos Específicos

1. **Desarrollar un sistema de notificaciones push para mantener a los usuarios informados**

Diseñar e implementar un sistema de notificaciones push que permita a la aplicación comunicarse de manera efectiva con los usuarios y profesionales. Este sistema se utilizará para recordar citas agendadas, notificar sobre cambios en horarios o servicios, y enviar promociones personalizadas. La tecnología de notificaciones push es clave para mejorar la interacción y el compromiso de los usuarios con la aplicación, además de garantizar que los profesionales estén al tanto de las reservas y modificaciones en sus agendas. Este módulo será optimizado para garantizar la entrega en tiempo real, adaptándose a las configuraciones de notificaciones del dispositivo del usuario.

1. **Implementar un módulo de configuración personalizada para los usuarios**

Desarrollar una funcionalidad que permita a los usuarios personalizar su experiencia dentro de la aplicación. Esto incluirá opciones como definir un rango de distancia máximo para la búsqueda de profesionales, guardar servicios o profesionales favoritos, y establecer recordatorios automáticos para la programación recurrente de servicios, como cortes de pelo o baños para mascotas. Este módulo incrementará la usabilidad de la plataforma y asegurará que cada usuario pueda ajustar la aplicación según sus necesidades individuales.

1. **Incorporar un sistema de reseñas y calificaciones para servicios y profesionales**

Crear un sistema robusto de evaluación que permita a los usuarios calificar y dejar reseñas sobre los servicios recibidos. Este módulo no solo fomentará la confianza entre los usuarios, sino que también servirá como una herramienta para que los profesionales reciban retroalimentación valiosa y mejoren sus servicios. Las calificaciones estarán visibles en los perfiles de los profesionales, ordenadas por relevancia y fecha, asegurando que los nuevos usuarios tengan información confiable al momento de elegir un servicio.

1. **Diseñar un sistema avanzado de filtros para la búsqueda de profesionales y servicios**

Implementar una funcionalidad que permita a los usuarios realizar búsquedas precisas de profesionales según criterios específicos, como precio, proximidad, calificaciones y especializaciones. Este módulo incluirá filtros dinámicos que se ajusten al contexto del usuario, ofreciendo resultados personalizados y facilitando la toma de decisiones. Los filtros avanzados serán optimizados para garantizar rapidez en las consultas y una experiencia fluida.

1. **Crear un panel administrativo para los profesionales registrados**

Desarrollar una sección dedicada dentro de la aplicación que permita a los profesionales gestionar su perfil y servicios de manera autónoma. Este panel incluirá herramientas para actualizar la información de sus servicios, configurar precios, gestionar su disponibilidad semanal y visualizar estadísticas sobre su rendimiento, como la cantidad de citas realizadas o ingresos estimados. Este módulo será diseñado con un enfoque intuitivo, permitiendo a los profesionales aprovechar al máximo la plataforma sin necesidad de conocimientos técnicos avanzados.

1. **Integrar la sincronización de horarios con calendarios externos**

Diseñar un sistema que permita a los profesionales sincronizar su agenda con plataformas de calendario externas como Google Calendar y Apple Calendar. Esto garantizará que los profesionales puedan gestionar su tiempo de manera efectiva, evitando conflictos entre citas agendadas dentro de la aplicación y otros compromisos externos. Este módulo se integrará mediante APIs confiables, asegurando actualizaciones en tiempo real y facilidad de uso.

1. **Optimizar la accesibilidad de la aplicación para todos los usuarios**

Implementar prácticas de diseño inclusivo que permitan a la aplicación ser utilizada por personas con diferentes capacidades. Esto incluirá la compatibilidad con lectores de pantalla para personas con discapacidades visuales, textos descriptivos alternativos para imágenes y un diseño de interfaz que sea intuitivo y claro. Además, se garantizará que los elementos interactivos sean accesibles mediante gestos o comandos de voz en dispositivos compatibles.

1. **Desarrollar un sistema de soporte al usuario dentro de la aplicación**

Implementar un módulo de soporte que brinde ayuda a los usuarios y profesionales en caso de dudas o inconvenientes. Este sistema incluirá una sección de preguntas frecuentes (FAQ), un formulario de contacto directo con el soporte técnico y la integración de un chatbot básico para responder consultas comunes de manera automática. Esto garantizará que los problemas se resuelvan de manera rápida, reduciendo la fricción en el uso de la aplicación.

1. **Gestionar cancelaciones y políticas de reprogramación de citas**

Diseñar un sistema que permita gestionar cancelaciones y reprogramaciones de manera eficiente, notificando a los usuarios y profesionales de cualquier cambio. Este módulo incluirá políticas personalizables para cada profesional, como límites de tiempo para cancelaciones gratuitas o penalizaciones en caso de cancelaciones tardías. Además, se optimizará la liberación de horarios para que puedan ser ocupados nuevamente por otros usuarios interesados.

1. **Incorporar reportes y estadísticas del uso de la aplicación**

Desarrollar una funcionalidad que genere reportes automáticos sobre el uso de la plataforma. Estos reportes incluirán datos como el número de usuarios activos, cantidad de reservas realizadas, servicios más populares y calificaciones promedio de los profesionales. Esta información será clave para tomar decisiones estratégicas sobre futuras mejoras de la aplicación y para que los profesionales puedan analizar su desempeño y ajustar sus servicios según las tendencias.

# Competencias del Perfil de Egreso

El desarrollo del proyecto EstilosExpress ha permitido aplicar y consolidar diversas competencias técnicas adquiridas durante la formación académica. Estas competencias están directamente relacionadas con las actividades clave del proyecto y han sido fundamentales para garantizar su correcto diseño, implementación y funcionalidad. A continuación, se destacan tres competencias principales que se aplicaron durante el desarrollo del proyecto:

**1.Diseño y desarrollo de aplicaciones móviles**

Esta competencia abarca la capacidad de diseñar, implementar y optimizar aplicaciones móviles funcionales, adaptadas a los estándares de usabilidad y requerimientos del usuario. En el proyecto EstilosExpress, se emplearon tecnologías modernas como Angular, Ionic y TypeScript para construir una aplicación móvil eficiente y escalable. Además, se implementaron principios de diseño intuitivo para garantizar una experiencia de usuario óptima.

**2.Gestión y administración de bases de datos relacionales**

La capacidad para diseñar, estructurar y administrar bases de datos es una competencia clave que se aplicó en el manejo de la información de usuarios, profesionales, servicios y reservas. En este proyecto, se utilizó Supabase, una solución basada en PostgreSQL, para gestionar la persistencia de datos de manera segura y escalable. Esto incluyó la creación de esquemas relacionales y consultas optimizadas para garantizar el correcto funcionamiento del sistema.

**3.Integración de tecnologías para el desarrollo de funcionalidades avanzadas**

Durante el desarrollo de la aplicación, se aplicaron competencias relacionadas con la integración de tecnologías para implementar funcionalidades avanzadas. Por ejemplo, el uso de Leaflet permitió incorporar geolocalización interactiva en tiempo real, mientras que la integración de la API de geolocalización del dispositivo móvil facilitó la búsqueda de profesionales cercanos. Estas capacidades técnicas permitieron resolver problemas complejos y mejorar la propuesta de valor de la aplicación.

**4.Implementación de medidas de seguridad en sistemas tecnológicos**  
La capacidad de integrar medidas de seguridad en el desarrollo de aplicaciones es una competencia esencial aplicada en el proyecto EstilosExpress. Esto incluye la implementación de cifrado mediante el hasheo de contraseñas para proteger la información de los usuarios y garantizar que los datos sensibles no sean visibles entre ellos. Estas medidas contribuyen a aumentar la confianza de los usuarios en la plataforma y aseguran el cumplimiento de estándares básicos de seguridad en aplicaciones tecnológicas.

**5.Aplicación de metodologías ágiles en el desarrollo de proyectos tecnológicos**  
La metodología Scrum fue utilizada durante el desarrollo de *EstilosExpress*, lo que permitió al equipo organizar el trabajo en ciclos iterativos y entregar funcionalidades incrementales. Esta competencia se refleja en la capacidad del equipo para planificar sprints, gestionar un backlog priorizado y realizar reuniones de seguimiento para resolver bloqueos y optimizar el flujo de trabajo. Este enfoque ágil ha sido clave para adaptar el desarrollo a los requerimientos del proyecto y garantizar entregables funcionales en cada etapa.

Estas competencias reflejan no solo el dominio técnico del equipo, sino también su capacidad para adaptarse a las demandas del mercado y a las expectativas del proyecto. Su correcta aplicación ha sido fundamental para el éxito de EstilosExpress en esta etapa de desarrollo.

# Acta de constitución

**Datos**

|  |  |
| --- | --- |
| Empresa / Organización | Equipo 3 Portafolio de titulo |
| Proyecto | EstiloExpress |
| Fecha de preparación | 2/10/2024 |
| Cliente | Duoc UC |
| Patrocinador principal |  |
| Gerente de Proyecto | Jorge Acuña |

**Descripción del Proyecto**

EstilosExpress es una aplicación móvil innovadora, diseñada para transformar la interacción entre clientes y profesionales independientes del sector de la peluquería, incluyendo peluqueros caninos, estilistas y barberos. La plataforma se posiciona como una herramienta esencial para aquellos clientes que buscan servicios de calidad, personalizados y cercanos a su ubicación. A través de su funcionalidad de geolocalización, los usuarios podrán localizar profesionales dentro de un rango específico de kilómetros, optimizando su tiempo y experiencia de búsqueda.

Además, la aplicación integra un sistema avanzado para la gestión de citas, permitiendo a los clientes reservar horarios disponibles de forma rápida y sin conflictos. Por su parte, los profesionales tendrán acceso a una agenda personalizada, donde podrán organizar su disponibilidad diaria y semanal, adaptándose a sus preferencias de trabajo, como atender en fines de semana o en horarios específicos. La personalización de perfiles es otra característica clave, permitiendo tanto a clientes como a profesionales mostrar fotografías, descripciones y datos relevantes que mejoren la interacción en la plataforma.

EstilosExpress no solo busca ofrecer una solución eficiente para las necesidades actuales, sino también establecer un estándar de calidad en la relación entre clientes y profesionales. En versiones futuras, se planea incorporar herramientas adicionales como la integración con aplicaciones de mensajería para facilitar la comunicación directa, así como la posibilidad de incluir nuevos idiomas que amplíen el alcance a mercados internacionales. Este proyecto se desarrolla con un enfoque en la simplicidad, la funcionalidad y la escalabilidad, garantizando que cada usuario obtenga una experiencia intuitiva y satisfactoria.

**Objetivos Generales y Específicos**

**Objetivo General**

Desarrollar una aplicación que facilite la interacción entre clientes y profesionales independientes del rubro de peluquería mediante herramientas de geolocalización, gestión de reservas y personalización de perfiles.

**Objetivos Específicos**

El proyecto EstilosExpress tiene como objetivo desarrollar una plataforma innovadora que sea intuitiva y accesible tanto en dispositivos móviles como en la web, brindando una experiencia eficiente y funcional para usuarios y profesionales. La aplicación integrará un sistema de búsqueda basado en geolocalización, que permitirá a los clientes localizar profesionales cercanos de manera rápida y precisa, optimizando así su tiempo y experiencia de búsqueda.

Además, se desarrollarán herramientas avanzadas para la gestión de citas, ofreciendo a los profesionales la posibilidad de administrar su disponibilidad diaria y semanal de forma dinámica, mientras que los clientes podrán reservar servicios de manera sencilla y sin conflictos de horarios. Otra característica clave será la personalización de perfiles, donde tanto clientes como profesionales podrán incluir fotografías y descripciones que destaquen sus intereses o especialidades, fomentando una conexión más personalizada y profesional entre ambas partes.

Por último, se garantizará que la plataforma cumpla con altos estándares de usabilidad y estabilidad, asegurando que cada funcionalidad opere de manera fluida y confiable. Esto no solo mejorará la experiencia del usuario, sino que también posicionará a EstilosExpress como una solución tecnológica confiable y adaptable a las necesidades cambiantes del mercado.

**Factores de Éxito del Proyecto**

El éxito del proyecto dependerá de los siguientes factores clave:

|  |  |
| --- | --- |
| **Factor de Éxito** | **Descripción** |
| **Finalización del producto mínimo viable** | Completar el desarrollo del MVP antes de la presentación académica, asegurando una base funcional sólida. |
| **Implementación de funcionalidades esenciales** | Lograr que las características clave del proyecto sean operativas y cumplan con un nivel alto de calidad. |
| **Retroalimentación positiva** | Obtener comentarios favorables de los usuarios durante las pruebas iniciales, lo que validará la experiencia de uso. |
| **Cumplimiento de plazos** | Respetar las fechas y entregables definidos en el cronograma para garantizar el avance ordenado del proyecto. |

**Fases del Proyecto con sus Principales Entregables**

|  |  |
| --- | --- |
| **Fase** | **Entregable Principal** |
| **Inicio del Proyecto** | Acta de constitución, plan inicial de trabajo. |
| **Diseño y Planificación** | Wireframes, mockups, especificaciones técnicas. |
| **Desarrollo del MVP** | Producto funcional con funcionalidades clave (geolocalización, reservas, perfiles). |
| **Pruebas y Ajustes** | Reporte de pruebas, correcciones de errores, mejoras de usabilidad. |
| **lanzamiento** | Versión final de la aplicación disponible para los usuarios. |

**Interesados Claves del Proyecto**

Los interesados clave en este proyecto incluyen:

|  |  |
| --- | --- |
| **Categoría** | **Detalles** |
| **Equipo de Desarrollo** | Jorge Acuña (Director del Proyecto), Nicolás Ayarza (Líder Técnico y Backend), Francois Mercidieu (Diseño y Frontend) |
| **Clientes Potenciales** | Personas que buscan servicios personalizados de peluquería |
| **Profesionales Potenciales** | Peluqueros caninos, estilistas y barberos independientes |
| **Entidad Supervisora** | Instituto Profesional DuocUC |

**Riesgos**

Los principales riesgos identificados para este proyecto son:

|  |  |
| --- | --- |
| **Riesgo Identificado** | **Descripción** |
| **Limitaciones de tiempo** | Completar el producto mínimo viable antes de la presentación académica. |
| **Dependencia de herramientas externas** | Uso de tecnologías de terceros como geolocalización y hosting, lo que podría generar limitaciones técnicas. |
| **Restricciones de recursos** | Equipo reducido de desarrollo y ausencia de financiamiento externo para herramientas o pruebas avanzadas. |
| **Aceptación de usuarios** | Riesgo de que los usuarios no adopten la aplicación como se espera debido a barreras técnicas o de usabilidad. |

**Hitos Principales**

| **Hito** | **Fecha** |
| --- | --- |
| **Inicio del Proyecto** | 22 de agosto de 2024 |
| **Finalización del Diseño y Planificación** | 05 de septiembre de 2024 |
| **Presentación MVP** | 28 de noviembre de 2024 |
| **Presentación académica** | 12 de diciembre de 2025 |
| **Lanzamiento Final** | 01 de junio de 2025 |

**Presupuesto**

No se ha definido un presupuesto formal para el proyecto debido a su naturaleza académica. Sin embargo, se estima que los costos se mantendrán bajos mediante el uso de herramientas gratuitas y recursos propios del equipo. Los gastos esperados incluyen licencias opcionales, servicios de hosting y herramientas de desarrollo.

**Requerimientos de Aprobación**

El proyecto requerirá la aprobación de los siguientes elementos para cada fase:

|  |  |
| --- | --- |
| **Fase del Proyecto** | **Descripción** |
| **Inicio del Proyecto** | Acta de constitución firmada y validada por el equipo y supervisores académicos. |
| **Diseño y Planificación** | Revisión y aprobación de los wireframes y especificaciones técnicas. |
| **Desarrollo del MVP** | Validación de pruebas de usuario y funcionalidad del MVP. |
| **Lanzamiento Final** | Revisión de calidad y conformidad por parte del equipo y los interesados claves. |

**Gerente o Líder del Proyecto**

El líder del proyecto es Jorge Acuña, quien será responsable de la gestión general, coordinación de tareas, y cumplimiento de los objetivos y plazos establecidos.

**Aprobaciones del Proyecto**

| **Aprobaciones** | | | |
| --- | --- | --- | --- |
| **Rol** | **Nombre** | **Firma** | **Fecha** |
| **Sponsor** | Duoc.uc | X | 28/11/2024 |
| **Cliente Principal** | Jonathan López | X | 28/11/2024 |
| **Jefe de Proyecto** | Jorge Acuña | X | 28/11/2024 |

# Asignación de Roles

En el desarrollo del proyecto EstilosExpress, la asignación de roles fue un elemento clave para garantizar una ejecución eficiente y organizada de las tareas, aprovechando las fortalezas individuales de cada integrante. A continuación, se describen los roles asignados y las principales responsabilidades asumidas por cada miembro del equipo:

**Jorge Acuña - Director del Proyecto**

Como director del proyecto, Jorge fue responsable de la planificación y supervisión de las actividades generales, asegurando el cumplimiento de los objetivos definidos en cada etapa. Su función principal consistió en coordinar al equipo mediante la metodología Scrum, organizando sprints y gestionando el backlog del proyecto. Además, lideró las reuniones de seguimiento, resolviendo bloqueos y asegurando que las tareas se realizaran dentro de los plazos establecidos. Jorge también se encargó de presentar los avances del proyecto a los supervisores y garantizar que el producto cumpliera con los estándares de calidad esperados.

**Nicolás Ayarza - Líder Técnico (Backend y Gestión de Datos)**

Nicolás desempeñó un rol fundamental como líder técnico, encargado del diseño e implementación de la arquitectura técnica del proyecto. Fue responsable de la configuración y gestión de la base de datos en Supabase, desarrollando funcionalidades clave como el registro de usuarios, el inicio de sesión y el sistema de reservas. También implementó medidas de seguridad, como el hasheo de contraseñas, para proteger la información de los usuarios. Su experiencia técnica fue esencial para optimizar las consultas a la base de datos y garantizar un rendimiento eficiente en el backend.

**Francoise Mercideus - Diseñador y Desarrollador Frontend (UI/UX)**

Francoise lideró el diseño de la interfaz de usuario y la implementación del frontend, asegurando que la aplicación ofreciera una experiencia visual atractiva y funcional. Se encargó de desarrollar pantallas clave utilizando Angular e Ionic, optimizando la aplicación para dispositivos móviles de diferentes tamaños. Además, integró el módulo de mapas interactivos con Leaflet, facilitando la funcionalidad de geolocalización. Francoise trabajó en estrecha colaboración con el líder técnico para garantizar una integración efectiva entre el frontend y el backend, además de realizar pruebas para asegurar la compatibilidad y el rendimiento de la interfaz.

En la siguiente página, se presenta la matriz de asignación de roles, que detalla de manera técnica y estructurada las responsabilidades específicas de cada integrante del equipo en el desarrollo del proyecto EstilosExpress. Esta matriz permite identificar claramente las funciones asignadas, las tareas realizadas y el nivel de participación de cada miembro en las distintas actividades del proyecto. Su elaboración responde a la necesidad de organizar y coordinar el trabajo de manera eficiente, garantizando que cada área crítica del proyecto esté adecuadamente cubierta y alineada con los objetivos establecidos.

**Matriz de Roles del Proyecto EstilosExpress**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tareas/Funciones** | **Jorge Acuña (Director del Proyecto)** | **Nicolás Ayarza (Líder Técnico)** | **Francoise Mercideus (Diseño y UI/UX)** |
| Planificación y seguimiento del proyecto | Responsable (R) | Consultado (C) | Informado (I) |
| Gestión del backlog (Scrum) | Responsable (R) | Consultado (C) | Consultado (C) |
| Diseño de la arquitectura técnica | Informado (I) | Responsable (R) | Consultado (C) |
| Configuración y gestión de la base de datos | Informado (I) | Responsable (R) | Consultado (C) |
| Desarrollo del backend | Informado (I) | Responsable (R) | Consultado (C) |
| Diseño de la interfaz de usuario | Informado (I) | Soporte (S) | Responsable (R) |
| Implementación del frontend | Consultado (C) | Soporte (S) | Responsable (R) |
| Integración frontend-backend | Consultado (C) | Responsable (R) | Responsable (R) |
| Implementación de geolocalización | Informado (I) | Responsable (R) | Soporte (S) |
| Pruebas y validaciones internas | Responsable (R) | Soporte (S) | Soporte (S) |
| Medidas de seguridad (hasheo de contraseñas, protección de datos) | Informado (I) | Responsable (R) | Consultado (C) |
| Presentación de avances del proyecto | Responsable (R) | Informado (I) | Informado (I) |

**Explicación de la Matriz**

* **Responsable (R):** Encargado principal de realizar la tarea y garantizar su cumplimiento.
* **Consultado (C):** Persona que aporta información o asesoramiento sobre la tarea.
* **Soporte (S):** Colabora en la realización de la tarea.
* **Informado (I):** Persona que recibe información sobre el progreso o resultados de la tarea, sin participación directa.

**¿Qué aporta esta Matriz de Roles al Proyecto?**

* **Organización:** Clarifica las responsabilidades de cada miembro, evitando confusiones o duplicación de esfuerzos.
* **Colaboración**: Establece la interdependencia entre los roles, promoviendo un trabajo en equipo eficiente.
* **Seguimiento:** Facilita la identificación de quién es responsable de cada tarea, mejorando el control del progreso del proyecto.

# Metodología Utilizada en el Proyecto

**Metodología Utilizada: Scrum**

El desarrollo del proyecto *EstilosExpress* ha seguido la metodología ágil **Scrum**, adaptándose a las necesidades específicas de cada etapa, desde la propuesta inicial hasta el desarrollo continuo previsto para junio de 2025. Este enfoque permitió trabajar de manera iterativa e incremental, priorizando la entrega de funcionalidades clave y asegurando la flexibilidad necesaria para incorporar mejoras basadas en retroalimentación constante.

**1. Etapa Inicial: Presentación de la Propuesta**

El proyecto comenzó con una etapa de investigación y planeación que se extendió durante un mes, previa al desarrollo técnico iniciado en octubre. En esta etapa se realizaron las siguientes actividades:

* **Identificación del Problema:** Se analizaron las necesidades del mercado, como el aumento en la tenencia de mascotas y la alta demanda de servicios de peluquería personalizada.
* **Definición de los Objetivos:** Se establecieron los objetivos generales y específicos del proyecto, así como los principales resultados esperados.
* **Esbozo del Alcance:** Se delinearon las funcionalidades clave para un Producto Mínimo Viable (MVP), priorizando el sistema de registro, la geolocalización y el sistema de reservas.

El resultado de esta etapa fue la aprobación de la propuesta por parte del equipo académico, lo que permitió dar inicio al desarrollo en octubre.

**2. Desarrollo del Proyecto**

Durante el desarrollo, se implementó la metodología Scrum en ciclos de trabajo de dos semanas (**sprints**), con objetivos específicos para cada iteración. Las principales actividades en esta etapa incluyeron:

**2.1. Organización Inicial**

* **Product Backlog:** Se definió una lista priorizada de funcionalidades necesarias para el MVP, como:
  + - Registro y autenticación de usuarios.
    - Módulo de geolocalización para mostrar profesionales cercanos.
    - Sistema de reservas con gestión de horarios.
* **Asignación de Roles:** Cada integrante del equipo asumió un rol específico:
  + - **Jorge Acuña:** Scrum Master y Director del Proyecto.
    - **Nicolás Ayarza:** Líder Técnico, encargado del backend y la base de datos.
    - **Francoise Mercideus:** Diseñador y Desarrollador **Frontend.**

**2.2. Ciclos de Sprints**

Los sprints se organizaron de la siguiente manera:

* **Sprint 1:** Configuración inicial del proyecto, incluyendo la estructura de la base de datos en Supabase y la implementación del sistema de registro con autenticación segura.
* **Sprint 2:** Desarrollo del módulo de geolocalización utilizando Leaflet, integrando la API de geolocalización del dispositivo móvil para identificar profesionales cercanos.
* **Sprint 3:** Implementación del sistema de reservas, con validación automática de horarios y notificaciones internas para confirmar citas.

**2.3. Pruebas y Validaciones**

En cada sprint, se realizaron pruebas internas de las funcionalidades desarrolladas, como:

* Verificación del correcto almacenamiento y recuperación de datos en la base de datos.
* Validación de la precisión del módulo de geolocalización y tiempos de carga.
* Pruebas del sistema de reservas para evitar conflictos de horarios.

**2.4. Revisión y Retrospectiva**

* En cada **Sprint Review**, el equipo presentó las funcionalidades desarrolladas, recolectó retroalimentación y realizó ajustes según fuera necesario.
* Durante las **Sprint Retrospectives**, se discutieron mejoras en la colaboración y optimización de tareas.

**3. Presentación del MVP**

El Producto Mínimo Viable (MVP) incluirá:

* Un sistema de registro y autenticación funcional.
* Módulo de geolocalización para mostrar profesionales cercanos.
* Sistema de reservas básico con gestión de horarios. Estas funcionalidades estarán listas para su demostración frente a los supervisores académicos el 12 de diciembre de 2024.

**4. Próximas etapas de desarrollo**

Tras la presentación del MVP, el desarrollo del proyecto continuará hasta junio de 2025 con el objetivo de incorporar funcionalidades avanzadas y optimizar la plataforma. Las actividades planeadas incluyen:

* **Integración de pagos en línea:** Incorporar métodos de pago seguros para reservas, proporcionando opciones confiables y accesibles para los usuarios.
* **Sistema de reseñas y calificaciones:** Permitir a los usuarios evaluar a los profesionales, mejorando la confianza y la transparencia en la prestación de servicios.
* **Optimización del rendimiento:** Mejorar los tiempos de carga y la fluidez de la aplicación, asegurando una experiencia de usuario óptima.
* **Documentación Técnica y Guías de Usuario:** Crear manuales que garanticen la usabilidad de la plataforma y faciliten su mantenimiento técnico.
* **Implementación de integración con aplicaciones de transporte:** Explorar la posibilidad de vincular la plataforma con aplicaciones de viajes como Uber, Cabify y DiDi, para que los usuarios puedan gestionar sus desplazamientos hacia el lugar donde se prestará el servicio, ampliando la funcionalidad y conveniencia de la aplicación.

El equipo mantendrá el enfoque iterativo de Scrum, organizando sprints para la implementación de cada funcionalidad, con retroalimentación continua que permita ajustar y mejorar el desarrollo en cada etapa.

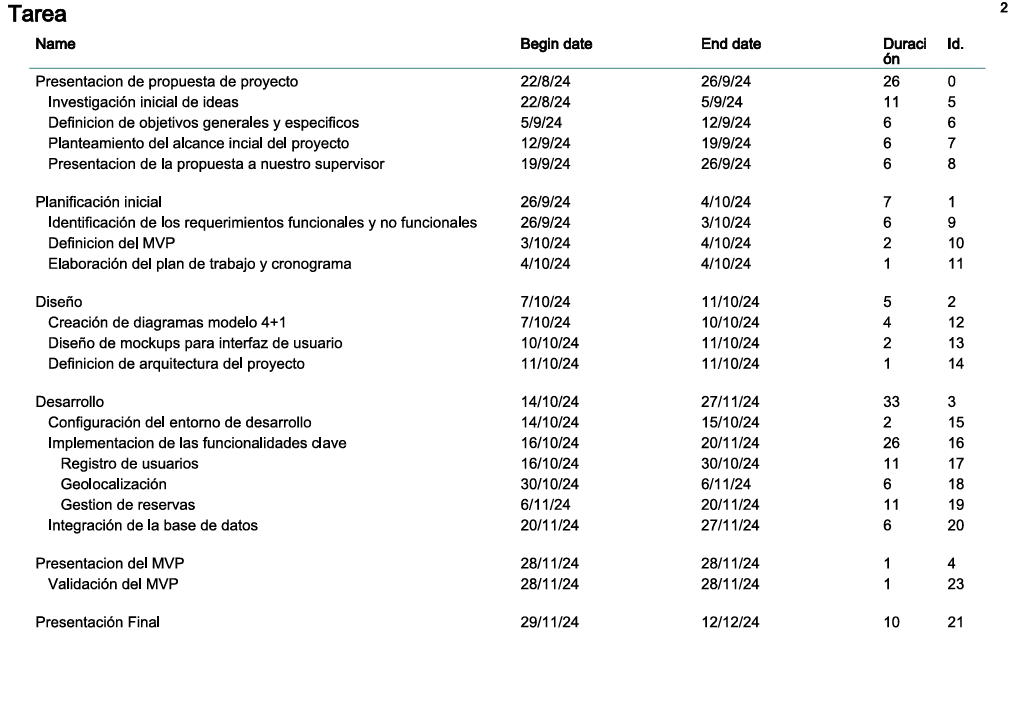
**Beneficios de Scrum en el Proyecto**

1. **Entrega Iterativa:** Permite entregar funcionalidades clave de forma gradual, asegurando que cada módulo cumpla con los requerimientos establecidos.
2. **Flexibilidad:** Facilita la incorporación de nuevas funcionalidades y ajustes en función de las necesidades del cliente y los usuarios finales.
3. **Colaboración:** La asignación clara de roles y las reuniones diarias aseguran una comunicación efectiva entre los integrantes del equipo.
4. **Calidad Garantizada:** Las revisiones y pruebas constantes minimizan errores y mejoran la estabilidad del producto.

**Conclusión**  
La implementación de Scrum ha sido clave para la organización y el avance continuo del proyecto *EstilosExpress*. Esta metodología no sólo permitió la entrega oportuna del MVP, sino que también sentó las bases para un desarrollo escalable y sostenible hasta junio de 2025, asegurando que la plataforma evolucione de acuerdo con las demandas del mercado.

# Carta Gantt

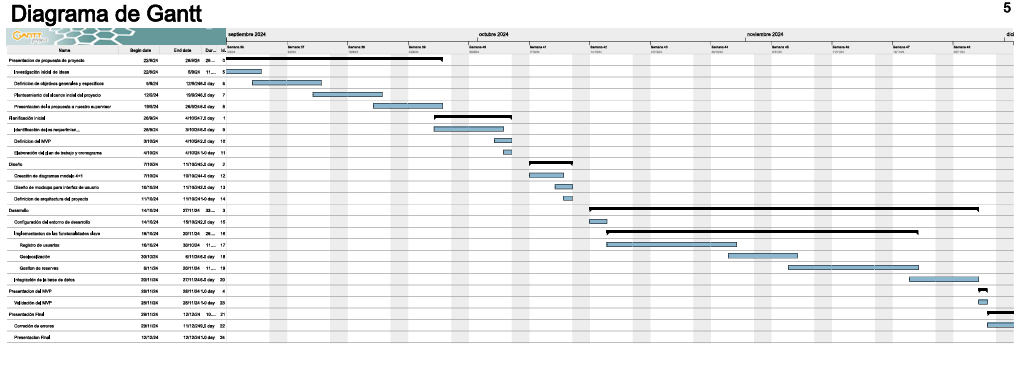
**Figura 1**: Planificación y organización de actividades mediante carta Gantt



**Nota**: Utilizamos la carta Gantt para organizar las actividades del proyecto, describiendo las tareas, asignando tiempos estimados para su realización y visualizando su secuencia.de manera clara.

**Fuente**: Elaboración propia.

**Figura 2**: Planificación y organización de actividades mediante carta Gantt



**Nota**: Esta forma de visualización de la carta Gantt facilita una mejor comprension de las actividades, sus dependencias y los tiempos asignados.

**Fuente**: Elaboración propia.

# Implementación del Proyecto

## ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE (ERS)

### 1. Introducción

La **Especificación de Requerimientos de Software (ERS)** para el proyecto EstilosExpress tiene como objetivo establecer una descripción detallada de las funcionalidades y características que debe cumplir el sistema. Este documento es fundamental para garantizar que todas las partes interesadas, incluidos los desarrolladores, clientes y profesionales, tengan una comprensión clara de los objetivos y expectativas del proyecto.

En este documento, se definen los requerimientos funcionales y no funcionales necesarios para implementar un sistema móvil moderno y eficiente. Además, se describen los actores principales que interactúan con la aplicación, las interfaces requeridas, las restricciones del sistema, y las dependencias externas que influencian su operación. La especificación también aborda aspectos clave como la usabilidad, el rendimiento y la seguridad, asegurando que el producto final cumpla con los estándares de calidad esperados.

El propósito de este documento no solo es servir como base para el diseño y desarrollo del software, sino también para garantizar que el sistema sea escalable, adaptable y capaz de evolucionar en futuras fases del proyecto. A través de un análisis exhaustivo, se detallan las funciones principales que ofrece la aplicación, así como las interacciones esperadas entre los usuarios, los profesionales, y las herramientas tecnológicas utilizadas.

Este documento es esencial para alinear los objetivos del equipo de desarrollo con las necesidades del mercado, proporcionando una guía clara y técnica que permita ejecutar el proyecto con éxito y preparar el camino para futuras expansiones.

### Propósito

El propósito de este documento es detallar los requerimientos funcionales y no funcionales del sistema EstilosExpress, una aplicación móvil destinada a conectar usuarios con profesionales independientes del sector de peluquería y estilismo. Este documento sirve como base para el desarrollo, asegurando que el sistema cumpla con las necesidades de los usuarios finales y los profesionales que ofrecen sus servicios. Además, proporciona una guía técnica que orienta al equipo de desarrollo durante la implementación, estableciendo las condiciones necesarias para un sistema escalable, seguro y funcional.

### 1.2 Ámbito del Sistema

El sistema EstilosExpress es una aplicación móvil diseñada para operar en dispositivos con sistemas operativos Android e iOS. Su principal objetivo es permitir a los usuarios buscar y reservar servicios de peluquería cercanos, mientras que los profesionales pueden gestionar sus horarios, servicios ofrecidos y recibir retroalimentación de los clientes. La aplicación utiliza tecnologías modernas, como geolocalización mediante Leaflet y bases de datos gestionadas con Supabase, para garantizar una experiencia intuitiva y fluida. El sistema está orientado a atender tanto a usuarios finales como a profesionales independientes.

### 1.3 Acrónimos y Abreviaturas

* MVP**:** Producto Mínimo Viable.
* UI**:** Interfaz de Usuario.
* UX**:** Experiencia de Usuario.
* API**:** Interfaz de Programación de Aplicaciones.
* CRUD**:** Crear, Leer, Actualizar y Eliminar.
* HTTPS**:** Protocolo Seguro de Transferencia de Hipertexto.
* JWT: Token Web JSON.

### 1.4 Visión General del Documento

Este documento está dividido en varias secciones que cubren todos los aspectos técnicos del sistema EstilosExpress. En la primera sección, se describe el propósito y alcance del sistema. La segunda sección aborda una descripción general de sus características, incluyendo las funcionalidades clave, los tipos de usuarios y las restricciones. La tercera sección presenta los requerimientos específicos, tanto funcionales como no funcionales, detallando los interfaces necesarios y otros aspectos técnicos relevantes. Este documento servirá como referencia técnica para el desarrollo y validación del sistema.

### 2. Descripción General

### 2.1 Perspectiva del Producto

El sistema EstiloExpress es una aplicación móvil diseñada para conectar usuarios con profesionales independientes del sector de peluquería y estilismo. Su propósito principal es facilitar la búsqueda, reserva y gestión de servicios de peluquería cercanos, proporcionando una experiencia intuitiva para usuarios finales como para profesionales.

La aplicación opera en dispositivos con sistema operativos android e iOS y emplea tecnologías de código abierto como ionic y angular para el desarrollo de la interfaz. También geolocalización mediante Leaflet, base de datos gestionada con Supabase, garantizando una plataforma segura, escalable y funcional.

Además, permite a los usuarios acceder a servicios personalizados y a los profesionales gestionar sus horarios, servicios ofrecidos y recibir retroalimentación de los clientes.

### 2.2 Funciones del Producto

El sistema ofrece las siguientes funcionalidades clave:

* Registro y Autenticación: Permite a los usuarios registrarse, iniciar sesión y recuperar contraseñas.
* Geolocalización: Muestra profesionales cercanos mediante mapas interactivos.
* Reserva de Servicios: Facilita el agendamiento de citas con selección de horarios y servicios específicos.
* Gestión de Perfiles: Usuarios y profesionales pueden editar sus datos y configurar sus preferencias.
* Calificaciones y Reseñas: Los clientes pueden evaluar y comentar sobre los servicios recibidos.

### 2.3 Características de los Usuarios

El sistema está diseñado para atender dos tipos principales de usuarios:

* Usuarios Clientes: Personas que buscan servicios de peluquería, barbería o peluquería canina cerca de su ubicación.
* Usuarios Profesionales: Peluqueros, estilistas y barberos independientes que ofrecen sus servicios a través de la plataforma.

### 2.4 Restricciones

El sistema requiere conexión a Internet para acceder a funcionalidades como geolocalización y reservas.

Actualmente, solo está disponible como aplicación móvil para Android e iOS.

Las reservas están sujetas a la disponibilidad configurada por los profesionales.

### Suposiciones y Dependencias

* Los usuarios finales poseen dispositivos móviles compatibles con Android 8.0 o superior, o iOS 13 o superior.
* Los profesionales gestionan activamente sus perfiles, incluyendo disponibilidad y servicios ofrecidos.
* Los servicios de Supabase y Leaflet API estarán disponibles para su integración continua.

### Requisitos Futuros

* Incorporación de pagos en línea para reservas.
* Expansión de funcionalidades para usuarios empresariales o locales pequeños.
* Desarrollo de una versión web de la aplicación.

### 3. Requisitos Específicos

### 3.1 Requisitos comunes de los interfaces

### Interfaces de usuario

La aplicación debe incluir:

* Pantallas para registro, inicio de sesión, y recuperación de contraseña.
* Un mapa interactivo para mostrar profesionales cercanos.
* Interfaz para reservar servicios con selección de horarios y confirmación.

### 3.1.2 Interfaces de hardware

* Compatible con dispositivos móviles con GPS habilitado.
* Resolución mínima de pantalla: 720x1280 píxeles.

### 3.1.3 Interfaces de software

* Comunicación con el backend mediante APIs REST.
* Integración con Supabase para autenticación y manejo de datos.
* Uso de Leaflet para la representación de mapas.

### 3.1.4 Interfaces de comunicación

* Protocolo HTTPS para garantizar la seguridad de las comunicaciones.
* Token JWT para autenticar solicitudes.

### 3.2 Requisitos Funcionales

| **Requerimiento** | **Descripción** |
| --- | --- |
| **Registro de Usuario** | Permitir a los usuarios crear una cuenta ingresando datos personales como nombre, correo y contraseña. |
| **Acceso a Cuenta Existente** | Facilitar el inicio de sesión para usuarios registrados mediante correo y contraseña. |
| **Modificar Perfil de Usuario** | Ofrecer la opción de editar información personal como nombre, foto de perfil, dirección, y contacto. |
| **Modificar y Eliminar Citas** | Permitir a los usuarios cambiar o cancelar citas previamente reservadas según las políticas del sistema. |
| **Gestionar Citas** | Habilitar a los profesionales para visualizar, aceptar o rechazar citas, y organizar su disponibilidad. |
| **Ganar y Canjear Puntos** | Implementar un sistema de recompensas que permita a los usuarios acumular puntos y canjearlos por beneficios. |
| **Reservar Citas con Profesionales** | Facilitar a los usuarios la selección de servicios, horarios y profesionales disponibles para agendar citas. |
| **Proveer Asistencia y Soporte** | Incluir un canal de atención para resolver dudas o problemas técnicos de los usuarios. |
| **Mostrar Profesionales Cercanos** | Utilizar geolocalización para listar profesionales en un radio cercano al usuario. |
| **Buscar Profesionales por Criterios Específicos** | Permitir filtrar profesionales por especialidades, calificaciones, precios y disponibilidad. |
| **Identificar Ubicación del Usuario** | Usar el GPS del dispositivo para determinar la ubicación del usuario y facilitar búsquedas cercanas. |
| **Compartir Experiencias y Reseñas en Redes Sociales** | Integrar la funcionalidad para compartir reseñas directamente en plataformas como Facebook o Instagram. |
| **Verificación de Reseñas de Usuarios Reales** | Garantizar que las reseñas sean realizadas solo por usuarios que hayan completado un servicio. |
| **Revisar Historial de Servicios** | Proveer un registro de servicios realizados con detalles como fecha, profesional y costo. |
| **Realizar Encuestas de Retroalimentación** | Enviar encuestas post-servicio para recopilar información sobre la satisfacción del usuario. |

**Nombre y descripción de casos de uso**

|  |  |
| --- | --- |
| Caso de Uso N°1 | Registro de Usuario |
| Actores | Usuario |
| Descripción | Este caso de uso permite a los nuevos usuarios registrarse en la aplicación proporcionando su información básica como nombre, correo electrónico, número de teléfono y contraseña. El sistema valida que los datos ingresados sean correctos y que el correo o número de teléfono no estén previamente registrados. |
| precondiciones | El usuario no debe tener una cuenta registrada previamente. |
| Postcondiciones | El usuario es registrado y se almacena la información en la base de datos. El usuario ahora puede iniciar sesión en la aplicación. |
| Flujo principal | 1. El usuario abre la aplicación y selecciona la opción de "Registrarse". 2. El sistema muestra un formulario para ingresar la información personal. 3. El usuario llena los campos requeridos. 4. El sistema valida los datos ingresados. 5. El sistema confirma que el usuario ha sido registrado correctamente. |
| Flujo alternativo | Si el correo o teléfono ya están registrados, el sistema muestra un mensaje de error y no permite continuar con el registro. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso N°2 | Acceso a Cuenta Existente |
| Actores | Usuario |
| Descripción | Permite a los usuarios ya registrados iniciar sesión en la aplicación utilizando su correo electrónico o número de teléfono y contraseña. |
| Precondiciones | El usuario debe estar registrado en la plataforma. |
| Postcondiciones | El usuario puede acceder a su cuenta y utilizar las funcionalidades protegidas. |
| Flujo principal | 1. El usuario selecciona la opción de "Iniciar sesión". 2. El sistema solicita correo electrónico o número de teléfono y contraseña. 3. El usuario ingresa sus credenciales. 4. El sistema valida las credenciales. 5. Si las credenciales son correctas, el usuario accede a su cuenta. |
| Flujo alternativo | Si las credenciales no son correctas, el sistema muestra un mensaje de error y solicita que se intente nuevamente. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso N°3 | Modificar Perfil de Usuario |
| Actores | Usuario |
| Descripción | Permite a los usuarios modificar su información personal, como nombre, correo electrónico, número de teléfono y contraseña. |
| Precondiciones | El usuario debe estar registrado e iniciado sesión. |
| Postcondiciones | La información del usuario es actualizada en la base de datos. |
| Flujo principal | 1. El usuario accede a su perfil y selecciona la opción de "Editar perfil". 2. El sistema muestra el formulario con los datos actuales del usuario. 3. El usuario modifica la información que desea cambiar. 4. El sistema valida los nuevos datos. 5. Los cambios son confirmados y almacenados en la base de datos. |
| Flujo alternativo | Si los nuevos datos son inválidos o ya están en uso, el sistema muestra un mensaje de error. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso N°4 | Modificar y Eliminar Citas |
| Actores | Usuario |
| Descripción | Permite a los usuarios modificar o eliminar citas previamente agendadas en la plataforma. |
| Precondiciones | El usuario debe tener citas reservadas en el sistema. |
| Postcondiciones | La cita es modificada o eliminada según la acción del usuario. |
| Flujo principal | 1. El usuario accede a su listado de citas agendadas. 2. Selecciona la cita que desea modificar o eliminar. 3. El sistema muestra las opciones para cambiar la fecha, hora o servicio, o eliminar la cita. 4. El usuario confirma la modificación o eliminación. 5. El sistema actualiza la cita en la base de datos. |
| Flujo alternativo | Si el usuario decide no realizar cambios, puede cancelar la operación y regresar al listado de citas. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso N°5 | Reservar Citas con Profesionales |
| Actores | Usuario |
| Descripción | Permite a los usuarios seleccionar un profesional y reservar una cita con él según su disponibilidad. |
| Precondiciones | El usuario debe estar registrado y haber iniciado sesión. |
| Postcondiciones | La cita es agendada con el profesional seleccionado y almacenada en la base de datos. |
| Flujo principal | 1. El usuario selecciona la opción de "Reservar cita". 2. El sistema muestra una lista de profesionales disponibles. 3. El usuario selecciona un profesional. 4. El sistema muestra las fechas y horas disponibles del profesional. 5. El usuario selecciona una fecha y hora. 6. El sistema confirma la reserva y guarda la cita en la base de datos. |
| Flujo alternativo | Si el profesional no tiene disponibilidad, el sistema sugiere otros horarios o profesionales. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso N°6 | Ganar y Canjear Puntos por Actividades o servicios |
| Actores | Usuario |
| Descripción | Este caso de uso permite a los usuarios ganar puntos al realizar actividades como, reservar citas o comprar servicios. Los puntos acumulados pueden ser canjeados por descuentos o beneficios dentro de la plataforma. |
| Precondiciones | El usuario debe estar registrado y haber realizado una transacción o actividad válida. |
| Postcondiciones | Los puntos son acumulados y registrados en la cuenta del usuario, y pueden ser canjeados por beneficios. |
| Flujo principal | 1. El usuario realiza una actividad válida (reservar cita, comprar servicio). 2. El sistema calcula los puntos correspondientes a la actividad. 3. El sistema suma los puntos al total de puntos del usuario. 4. El usuario accede a la sección de canje de puntos. 5. El usuario selecciona un beneficio o descuento y lo canjea. |
| Flujo alternativo | Si el usuario no tiene suficientes puntos, el sistema muestra un mensaje informando de la insuficiencia de puntos. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso N°7 | Reservar Citas con Profesionales Específicos |
| Actores | Usuario |
| Descripción | Permite a los usuarios buscar y reservar citas con profesionales específicos según su disponibilidad. |
| Precondiciones | El usuario debe estar registrado e iniciado sesión. |
| Postcondiciones | La cita es agendada con el profesional seleccionado y se guarda en la base de datos. |
| Flujo principal | 1. El usuario selecciona la opción de "Buscar profesional". 2. El sistema muestra una lista de profesionales disponibles. 3. El usuario selecciona un profesional específico. 4. El sistema muestra las fechas y horas disponibles. 5. El usuario selecciona una fecha y hora. 6. El sistema confirma la reserva y guarda la cita en la base de datos. |
| Flujo alternativo | Si el profesional no tiene disponibilidad, el sistema sugiere otros horarios o profesionales. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso N°8 | Mostrar Profesionales Cercanos |
| Actores | Usuario |
| Descripción | Permite a los usuarios visualizar una lista de profesionales cercanos basados en su ubicación geográfica. |
| Precondiciones | El usuario debe permitir que la aplicación acceda a su ubicación. |
| Postcondiciones | El sistema muestra los profesionales más cercanos, permitiendo al usuario seleccionar y reservar citas. |
| Flujo principal | 1. El usuario accede a la opción de "Ver profesionales cercanos". 2. El sistema solicita acceso a la ubicación del usuario. 3. El usuario concede el permiso de ubicación. 4. El sistema muestra una lista de profesionales cercanos ordenados por distancia. 5. El usuario selecciona un profesional para ver más detalles o reservar una cita. |
| Flujo alternativo | Si el usuario no concede el acceso a la ubicación, el sistema muestra un mensaje indicando que la funcionalidad no estará disponible. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso N°9 | Buscar Profesionales por Criterios Específicos |
| Actores | Usuario |
| Descripción | Este caso de uso permite a los usuarios buscar profesionales utilizando criterios como especialidad, calificación, y ubicación. |
| Precondiciones | El usuario debe estar registrado y haber iniciado sesión. |
| Postcondiciones | El sistema filtra y muestra una lista de profesionales que cumplen con los criterios ingresados. |
| Flujo principal | 1. El usuario selecciona la opción de "Buscar profesional". 2. El sistema muestra un formulario con criterios de búsqueda (especialidad, ubicación, calificación, etc.). 3. El usuario ingresa los criterios deseados. 4. El sistema filtra y muestra los resultados basados en los criterios. 5. El usuario selecciona un profesional de la lista para ver más detalles o reservar una cita. |
| Flujo alternativo | Si no se encuentran resultados, el sistema muestra un mensaje sugiriendo ajustar los criterios de búsqueda. |

|  |  |
| --- | --- |
| Caso de Uso N°10 | Identificar Ubicación del Usuario para Mostrar Opciones Cercanas |
| Actores | Usuario |
| Descripción | Permite al sistema identificar la ubicación actual del usuario para ofrecerle opciones cercanas, como profesionales o servicios. |
| Precondiciones | El usuario debe permitir el acceso a su ubicación. |
| Postcondiciones | El sistema utiliza la ubicación del usuario para mostrar recomendaciones cercanas. |
| Flujo principal | 1. El usuario accede a una funcionalidad que requiere información de ubicación. 2. El sistema solicita el permiso para acceder a la ubicación. 3. El usuario otorga el permiso. 4. El sistema identifica la ubicación y muestra resultados personalizados. |
| Flujo alternativo | Si el usuario no concede el permiso de ubicación, el sistema muestra un mensaje y no puede ofrecer resultados basados en la ubicación. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso N°11 | Compartir Experiencias y Reseñas en Redes Sociales |
| Actores | Usuario |
| Descripción | Permite a los usuarios compartir reseñas y experiencias en redes sociales directamente desde la plataforma. |
| Precondiciones | El usuario debe haber utilizado un servicio en la plataforma y tener una cuenta en la red social en la que desea compartir. |
| Postcondiciones | La reseña o experiencia es compartida en la red social seleccionada por el usuario. |
| Flujo principal | 1. El usuario selecciona una cita o servicio completado. 2. El sistema ofrece la opción de redactar una reseña o experiencia. 3. El usuario redacta la reseña y selecciona la opción para compartir en redes sociales. 4. El sistema publica la reseña en la red social seleccionada. |
| Flujo alternativo | Si el usuario cancela, la reseña no es publicada. |

|  |  |
| --- | --- |
| Caso de Uso N°12 | Verificación de Reseñas de Usuarios Reales |
| Actores | Usuario, sistema |
| Descripción | Este caso de uso permite verificar que las reseñas publicadas en la plataforma provienen de usuarios reales que han utilizado los servicios, vinculando las reseñas a transacciones o interacciones verificadas. |
| Precondiciones | El usuario debe haber reservado y completado un servicio. |
| Postcondiciones | La reseña es publicada y marcada como "verificada", asegurando su autenticidad. |
| Flujo principal | 1. El usuario completa un servicio y escribe una reseña. 2. El sistema verifica que el usuario ha utilizado el servicio relacionado a la reseña. 3. El sistema etiqueta la reseña como "verificada" y la publica en el perfil del profesional. |
| Flujo alternativo | Si el usuario no ha completado el servicio o no es un usuario real, el sistema bloquea la publicación de la reseña. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso N°13 | Revisar Historial de Servicios Prestados |
| Actores | Usuario, Profesional |
| Descripción | Este caso de uso permite a los usuarios y profesionales revisar el historial de servicios prestados o recibidos a través de la plataforma. |
| Precondiciones | El usuario o profesional debe haber utilizado o brindado servicios a través de la plataforma. |
| Postcondiciones | Se muestra un historial detallado de los servicios completados, incluyendo fechas, tipos de servicios, y profesionales involucrados. |
| Flujo principal | 1. El usuario o profesional accede a la sección de historial de servicios. 2. El sistema muestra una lista de los servicios prestados o recibidos. 3. El usuario o profesional puede filtrar y revisar los detalles de cada servicio. |
| Flujo alternativo | Si no hay servicios en el historial, el sistema muestra un mensaje indicando que no se han registrado servicios. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso N°14 | Realizar Encuestas para Obtener Retroalimentación |
| Actores | Usuario, Sistema |
| Descripción | Este caso de uso permite al sistema administrar encuestas para recopilar retroalimentación de los usuarios sobre la plataforma y los servicios ofrecidos. |
| Precondiciones | El usuario debe estar registrado y haber utilizado la plataforma o sus servicios. |
| Postcondiciones | Las respuestas de la encuesta se registran y almacenan para su análisis. |
| Flujo principal | 1. El sistema invita al usuario a participar en una encuesta después de utilizar un servicio o al azar. 2. El usuario accede a la encuesta y responde las preguntas. 3. El sistema registra las respuestas y las almacena para análisis futuro. 4. El sistema puede ofrecer un incentivo (como puntos) al usuario por completar la encuesta. |
| Flujo alternativo | El usuario puede declinar la participación en la encuesta. |
|  |  |

### 3.3 Requisitos No Funcionales Rendimiento:

* El sistema debe garantizar un tiempo de respuesta inferior a 2 segundos para la mayoría de las operaciones críticas, como el registro de usuarios, la autenticación, la consulta de profesionales cercanos y la reserva de servicios.

**Escalabilidad:**

* La arquitectura del sistema debe permitir el crecimiento de la base de usuarios, soportando hasta 10,000 usuarios concurrentes sin comprometer el rendimiento. Esto incluye la capacidad de integrar nuevas funcionalidades y adaptarse a mayores volúmenes de datos en el futuro.

**Disponibilidad:**

* El sistema debe estar disponible al menos el 99.9% del tiempo, minimizando los períodos de inactividad programada y no programada. Esto incluye un diseño resiliente que permita la recuperación ante fallos en servicios externos o infraestructuras.

**Seguridad:**

* Las contraseñas de los usuarios deben ser encriptadas utilizando algoritmos como bcryptjs para evitar que se almacenen en texto plano. Además, todas las comunicaciones entre el frontend y el backend deben realizarse a través del protocolo HTTPS, garantizando la confidencialidad y la integridad de los datos.

**Compatibilidad:**

* La aplicación debe ser compatible con dispositivos móviles que utilicen Android (8.0 o superior) e iOS (13 o superior). Debe adaptarse correctamente a diferentes resoluciones de pantalla, con un diseño responsive que proporcione una experiencia uniforme en todos los dispositivos compatibles.

**Usabilidad:**

* La interfaz de usuario debe ser intuitiva, con un diseño limpio y adaptado a dispositivos móviles. Los usuarios deben ser capaces de completar tareas clave, como el registro, la búsqueda de profesionales y la reserva de citas, en menos de 5 pasos.

**Mantenibilidad:**

* El código fuente del sistema debe ser modular y documentado adecuadamente, permitiendo que futuros desarrolladores puedan realizar actualizaciones y mantenimiento sin dificultades. Además, se deben utilizar herramientas estándar como control de versiones (Git) para gestionar el ciclo de desarrollo.

**Portabilidad:**

* Aunque inicialmente el sistema está diseñado como una aplicación móvil, su diseño debe permitir su futura implementación en plataformas web con cambios mínimos en la arquitectura del backend.

**Interoperabilidad:**

* El sistema debe ser capaz de integrarse con servicios externos, como APIs de OSRM, redes sociales y sistemas de pago en línea, de manera eficiente y segura.

**Accesibilidad:**

* La aplicación debe cumplir con estándares básicos de accesibilidad, como compatibilidad con lectores de pantalla y textos descriptivos, para garantizar que usuarios con discapacidades puedan interactuar con el sistema.

### 3.4 Otros Requisitos

* Encriptación de contraseñas con Bcryptjs.
* Compatibilidad con resoluciones de pantalla superiores a 5 pulgadas.
* Soporte para múltiples idiomas en futuras expansiones.

## Modelo 4+1

La implementación del proyecto EstilosExpress consistió en desarrollar una aplicación móvil funcional que cumpliera con los requerimientos definidos en las etapas de planificación y diseño. Para garantizar una ejecución técnica clara y escalable, se utilizó el modelo de **arquitectura 4+1**, que permitió estructurar la solución en cinco vistas complementarias: lógica, desarrollo, procesos, física y casos de uso. Este modelo fue clave para organizar las funcionalidades del sistema, optimizar su desarrollo y comunicar la arquitectura a diferentes audiencias técnicas y no técnicas.

El desarrollo se llevó a cabo utilizando un enfoque iterativo basado en la metodología Scrum, asegurando la entrega incremental de funcionalidades en ciclos de dos semanas. Las tecnologías utilizadas incluyeron Angular e Ionic para el frontend, Supabase como backend para la gestión de datos y autenticación, y Leaflet para la integración de mapas interactivos. Estas herramientas fueron seleccionadas por su capacidad de proporcionar una solución robusta, eficiente y adaptada al entorno móvil.

A continuación, se describe en detalle cómo se implementó cada vista del modelo 4+1, junto con los diagramas correspondientes que ilustran los componentes principales, la interacción entre ellos y los casos de uso del sistema.

**Vista Lógica**

La **Vista Lógica** del proyecto *EstilosExpress* describe la estructura funcional del sistema, identificando los principales módulos que componen la aplicación y cómo interactúan entre sí para proporcionar las funcionalidades requeridas. Esta vista se enfoca en representar el diseño lógico de los componentes y sus relaciones, lo que permite entender el flujo de datos y las responsabilidades de cada módulo dentro del sistema.

El diseño lógico del sistema está organizado en módulos clave, cada uno encargado de una funcionalidad específica, asegurando modularidad, escalabilidad y mantenimiento sencillo. Los módulos principales incluyen:

1. **Módulo de Registro y Autenticación:**  
   Responsable de la creación y gestión de cuentas de usuarios y profesionales, asegurando la autenticación segura mediante el uso de tokens y cifrado de contraseñas.
2. **Módulo de Geolocalización:**  
   Permite a los usuarios localizar profesionales cercanos en un mapa interactivo, utilizando coordenadas GPS obtenidas en tiempo real.
3. **Módulo de Reservas:**  
   Gestiona las citas entre usuarios y profesionales, incluyendo la validación de horarios disponibles, confirmaciones y notificaciones.
4. **Módulo de Perfiles:**  
   Facilita la personalización de los perfiles de los usuarios y profesionales, permitiendo la actualización de datos personales y la visualización de servicios ofrecidos.
5. **Backend y Base de Datos:**  
   Gestionado a través de Supabase, es responsable de almacenar, consultar y asegurar la integridad de los datos, como registros de usuarios, profesionales, reservas y servicios.

La interacción entre estos módulos asegura una experiencia fluida para los usuarios, desde la búsqueda de servicios hasta la confirmación de reservas. El diagrama a continuación ilustra cómo se organizan y relacionan estos componentes dentro del sistema.

**Diagrama de Componentes**

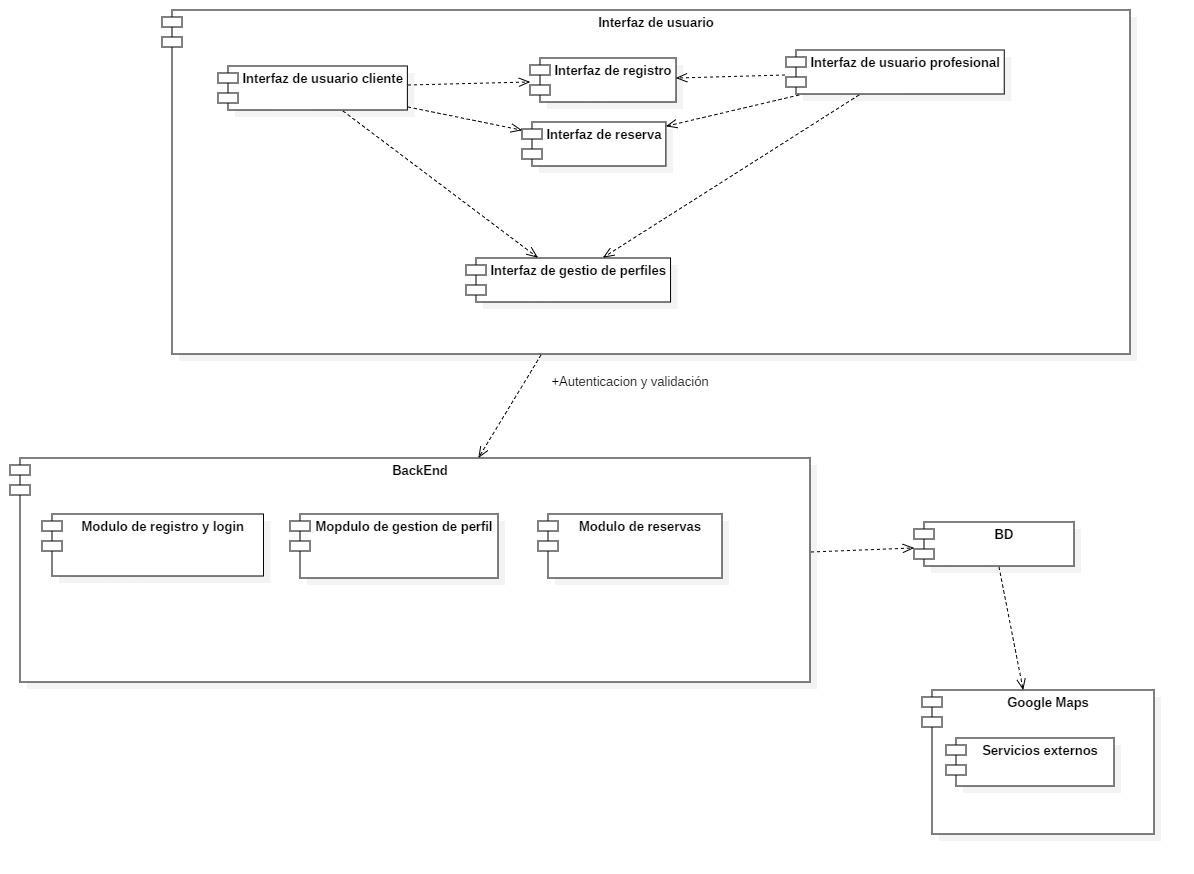
El diagrama de componentes muestra cómo los diferentes módulos de la aplicación interactúan entre sí y con servicios externos. Esta representación incluye tanto los elementos del frontend (interfaz de usuario) como del backend (servidor), además de las conexiones con servicios como Google Maps y la base de datos.

**Explicación del Diagrama de Componentes**

* **Frontend (Interfaz de Usuario):**
  + Incluye las interfaces de registro, gestión de reservas y perfiles, tanto para usuarios clientes como para profesionales.
  + Estas interfaces se comunican con el backend para obtener y enviar datos.
* **Backend:**
  + Compuesto por tres módulos principales:
    - **Módulo de Registro y Login:** Gestiona la autenticación de usuarios.
    - **Módulo de Gestión de Perfiles:** Administra la información de los usuarios y profesionales.
    - **Módulo de Reservas:** Permite gestionar las citas entre usuarios y profesionales.
  + Los módulos se conectan a la base de datos para almacenar y recuperar información.
* **Servicios Externos:**
  + Uso de OSRM para la geolocalización y cálculo de rutas.
  + La base de datos almacena información crítica como usuarios, servicios, y reservas.

A continuación, se presenta el diagrama de componentes que ilustra la estructura modular del sistema y sus interacciones:

**Figura 3:** Diagrama de componentes



**Nota:** Se ilustra la estructura modular del sistema, destacando la interacción entre los componentes del frontend y el backend, así como conexiones con servicios externos.

**Fuente:** Elaboración propia.

**Diagrama de Clases**

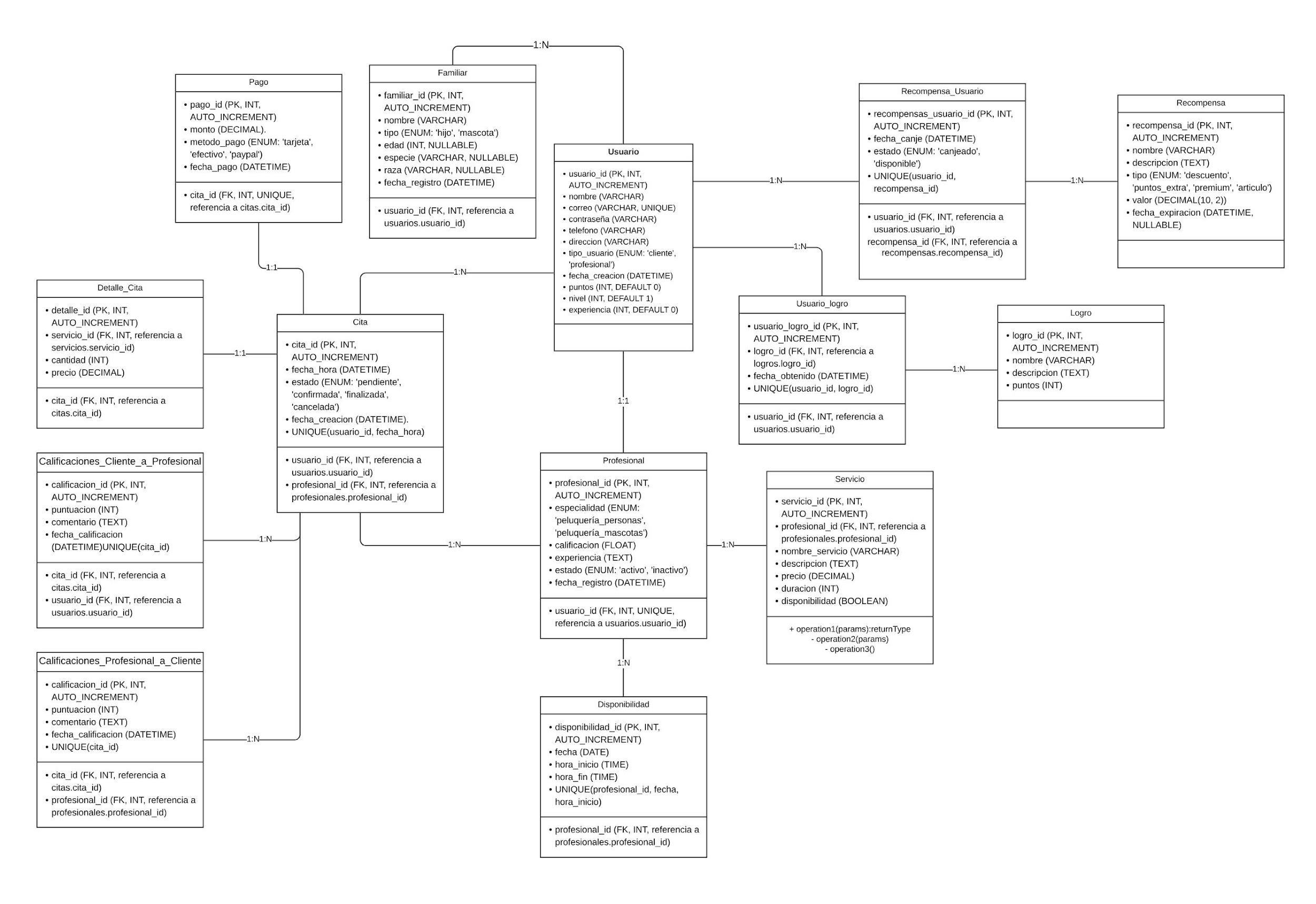
El diagrama de clases detalla las entidades principales del sistema, sus atributos, métodos y las relaciones entre ellas. Esto proporciona una visión más técnica y detallada de la estructura interna del sistema.

**Explicación del Diagrama de Clases**

* **Entidades Principales:**
  + **Usuario:** Representa tanto a clientes como a profesionales, con atributos como nombre, correo y tipo de usuario.
  + **Profesional:** Subtipo de usuario que incluye atributos específicos como especialidades y servicios ofrecidos.
  + **Cita:** Relaciona a los usuarios con los profesionales, gestionando horarios y estados de las reservas.
  + **Servicios:** Define los servicios ofrecidos por los profesionales, incluyendo precios y duración.
  + **Calificaciones:** Permite a los usuarios dejar reseñas sobre los servicios prestados.
* **Relaciones:**
  + Las asociaciones entre las entidades muestran cómo interactúan entre sí. Por ejemplo, un usuario puede tener varias reservas (relación 1:N), y un profesional puede ofrecer múltiples servicios.

A continuación, se presenta el diagrama de clases que detalla la estructura lógica interna del sistema:

**Figura 4:** Diagrama de clases



**Nota:** El diagrama muestra todas las principales entidades del sistema, permite visualizar claramente sus atributos, métodos y la relación entre ellas.

**Fuente:** Elaboración propia.

**Diagrama de Paquetes**

El **Diagrama de Paquetes** es una representación lógica que muestra cómo se organiza el sistema en agrupaciones (paquetes) que reflejan la estructura modular del código. Este diagrama es útil para entender la organización del sistema desde el punto de vista del desarrollo, facilitando el mantenimiento y la escalabilidad.

En el caso de *EstilosExpress*, el sistema está dividido en paquetes que corresponden a las principales funcionalidades de la aplicación y su interacción con servicios externos. Cada paquete incluye submódulos o componentes específicos que cumplen roles definidos dentro del proyecto.

**Estructura del Diagrama de Paquetes**

* + **Frontend:**
    - Representa la interfaz de usuario de la aplicación móvil.
  + Incluye submódulos como:
    - **Registro:** Para gestionar el registro y autenticación de usuarios.
    - **Reservas:** Para permitir a los usuarios agendar citas con los profesionales.
    - **Perfiles:** Para visualizar y editar los datos personales.

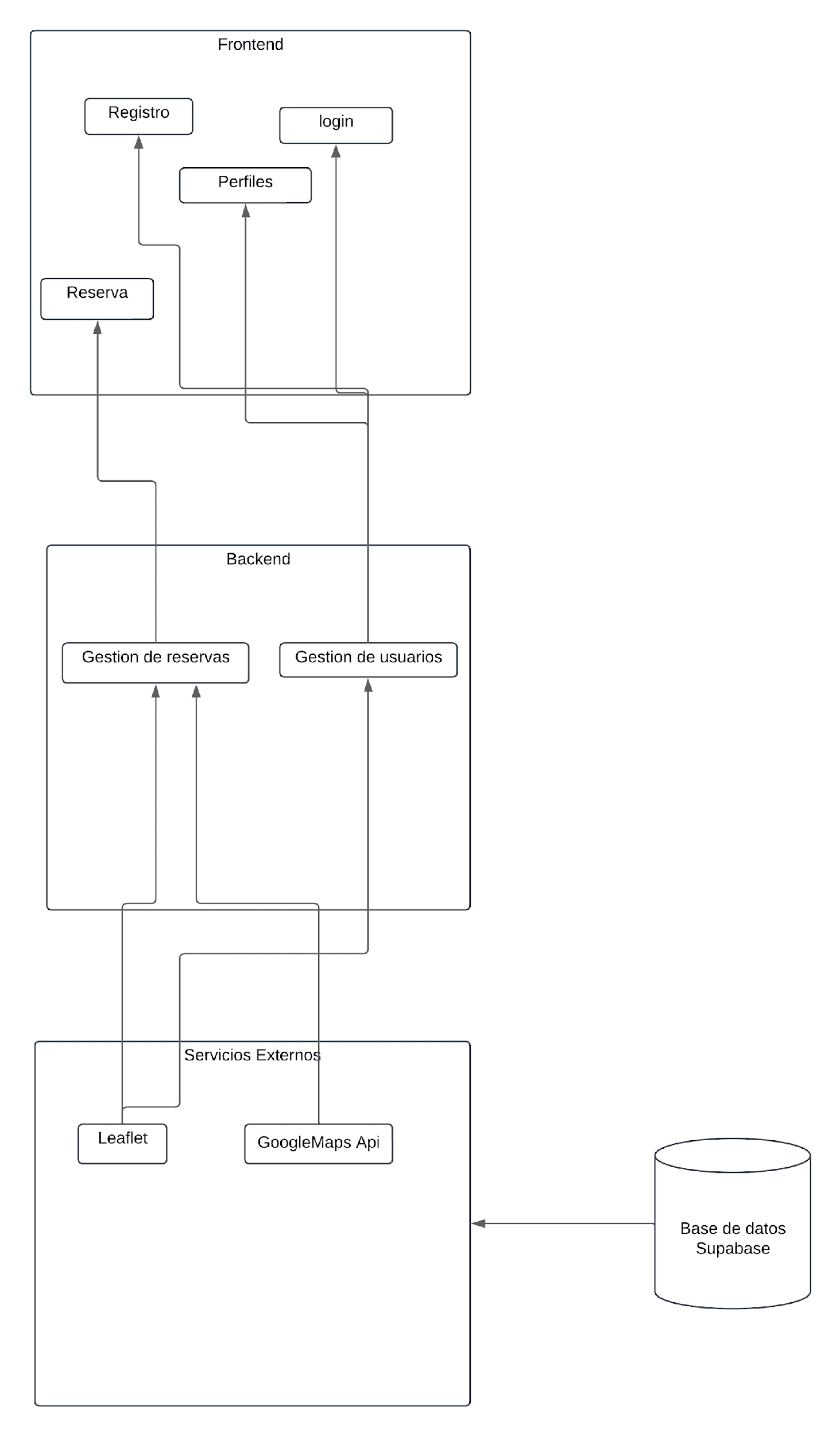
1. **Backend:**
   * Se encarga de procesar las solicitudes del frontend y gestionar la lógica del negocio.
   * Incluye submódulos como:
     + **Gestión de Usuarios:** Para manejar los datos de usuarios y profesionales.
     + **Gestión de Reservas:** Para procesar y almacenar información de citas.
     + **Gestión de Perfiles:** Para sincronizar y actualizar la información del usuario.
2. **Servicios Externos:**
   * Incluye integraciones con herramientas y APIs externas como:
     + **Leaflet:** Para la visualización de mapas interactivos.
     + **OSRM API:** Para calcular rutas y navegación.
     + **Nominatim API:** Para transformar dirección ingresada a coordenadas y viceversa.
3. **Base de Datos:**
   * Es el paquete donde se gestionan todas las operaciones relacionadas con el almacenamiento de datos, como usuarios, reservas, servicios, y perfiles.

**Relaciones entre Paquetes**

* El **Frontend** se comunica con el **Backend** para enviar y recibir datos.
* El **Backend** interactúa con la **Base de Datos** para almacenar y recuperar información crítica.
* Los **Servicios Externos** proporcionan funcionalidades adicionales, como geolocalización y cálculo de rutas, a través de APIs conectadas al backend.

A continuación, se presenta el diagrama de paquetes que ilustra la organización lógica de los módulos del sistema y sus interacciones:

**Figura 5:** Diagrama de paquete



**Nota:** Se muestra la organización del sistema en módulos principales, detallando cómo agrupan las funcionalidades del frontend, backend, servicios externos y base de datos, y cómo interactúan entre sí.

**Fuente:** Elaboración propia.

**Vista de Procesos**

La Vista de Procesos describe cómo los diferentes componentes del sistema interactúan en tiempo de ejecución para cumplir con las principales funcionalidades de EstilosExpress. Esta vista se enfoca en los flujos de datos, las interacciones entre el frontend, backend, y servicios externos, y los pasos necesarios para ejecutar casos de uso clave. A continuación, se detallan los procesos implementados más relevantes: el registro de usuarios y la reserva de servicios.

**1. Flujo de Registro de Usuario**

El proceso de registro permite a los nuevos usuarios crear una cuenta en la aplicación de forma segura y eficiente. Este flujo interactúa con el frontend, el backend, y la base de datos para validar y almacenar la información ingresada.

**Pasos del Proceso:**

1. El usuario ingresa sus datos en el formulario de registro desde la aplicación móvil.
2. El frontend valida los datos básicos localmente y los envía al backend.
3. El backend realiza validaciones adicionales, como comprobar si el correo electrónico ya está registrado.
4. Si los datos son válidos, se almacena la información del usuario en la base de datos.
5. El backend genera un token de autenticación y envía una respuesta de éxito al frontend.
6. El frontend muestra al usuario un mensaje confirmando el registro exitoso.

Este flujo asegura que los datos de los usuarios sean gestionados de manera segura, implementando validaciones tanto en el cliente como en el servidor.

**2. Flujo de Reserva de Servicio**

El proceso de reserva permite a los usuarios agendar servicios con los profesionales disponibles, asegurando la sincronización de horarios y la actualización en tiempo real de la base de datos.

**Pasos del Proceso:**

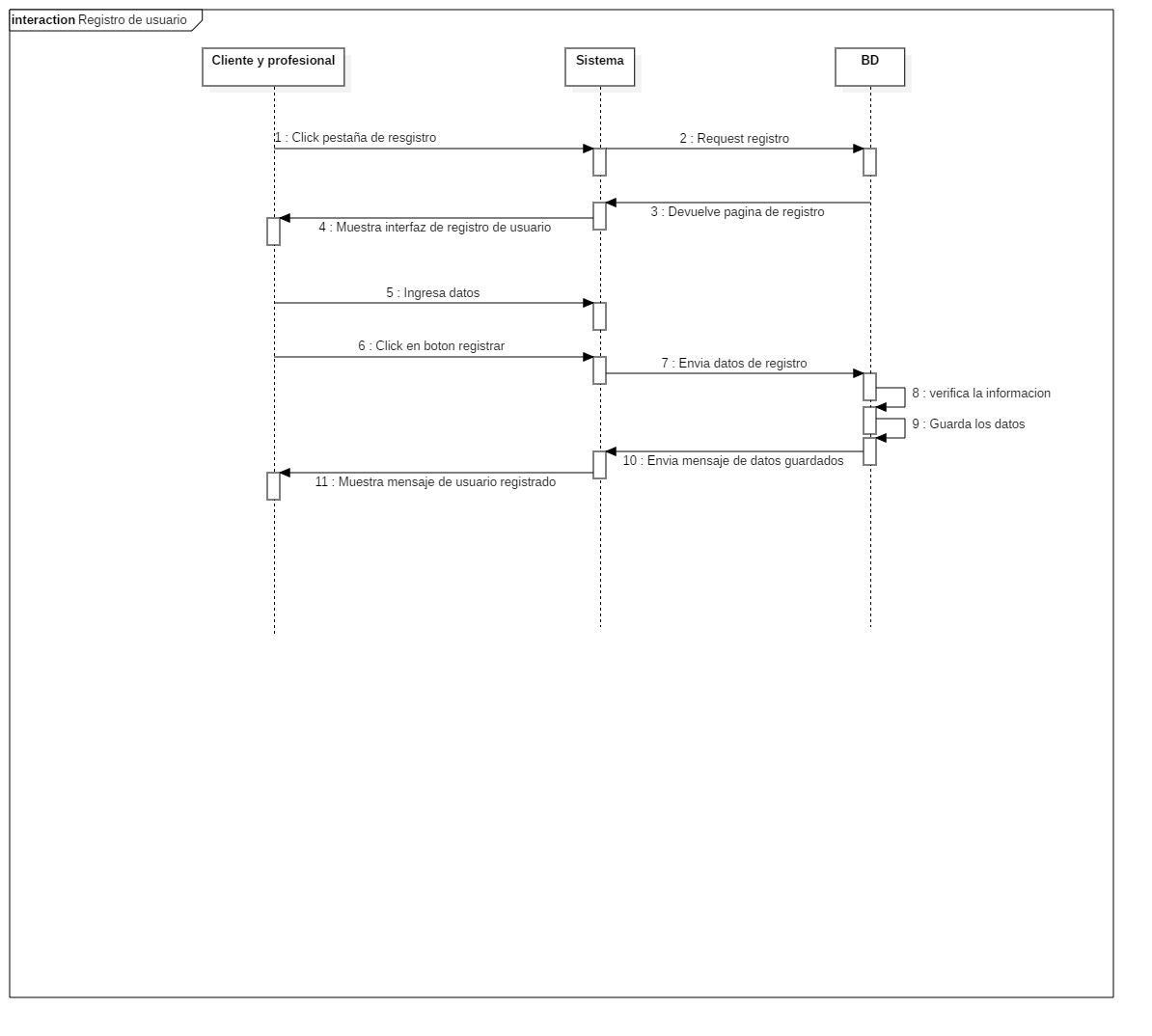
1. El usuario selecciona un profesional y un servicio desde la aplicación móvil.
2. El frontend envía una solicitud al backend para consultar la disponibilidad del horario seleccionado.
3. El backend verifica en la base de datos si el horario está libre.
4. Si está disponible, el backend guarda la reserva en la base de datos y actualiza el calendario del profesional.
5. El backend envía una confirmación al frontend.
6. El frontend muestra al usuario un mensaje confirmando la reserva exitosa.

Este proceso garantiza que los horarios no se solapen y que tanto el usuario como el profesional estén informados en tiempo real.

**Diagramas de Secuencia**

A continuación, se presentan los diagramas que ilustran los flujos descritos:

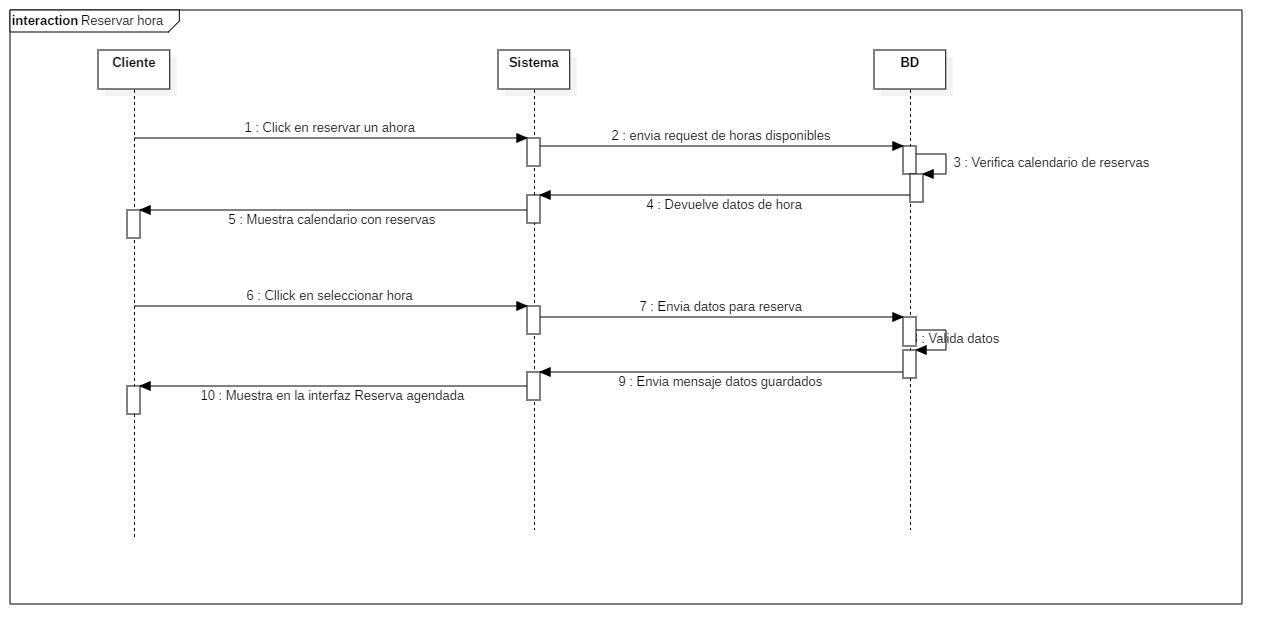
**Figura 6:** Diagrama de flujo para registro de usuario



**Nota:** Se muestran los pasos del flujo para realizar el registro de usuario, mostrando como frontend, backend y base de datos interactúan para validar, almacenar y confirmar la creación de la cuenta.

**Fuente:** Elaboración propia.

**Figura 7:** Diagrama de flujo para reserva de hora



**Nota:** Se muestran los pasos del flujo de reserva de servicio, mostrando como frontend, backend y base de datos interactúan para consultar disponibilidad, confirmar la reserva y actualizar los calendarios en tiempo real.

**Fuente:** Elaboración propia.

**Vista Física**

La **Vista Física** detalla cómo se distribuyen los diferentes componentes del sistema en el entorno de despliegue. Esta vista es crucial para entender la infraestructura técnica que soporta la aplicación *EstilosExpress*, mostrando los nodos físicos o virtuales y sus interacciones.

**Estructura de la Infraestructura**

1. **Cliente Móvil:**
   * Representa la aplicación instalada en dispositivos móviles, utilizada por los usuarios y profesionales para interactuar con el sistema.
   * Se comunica con el servidor a través de solicitudes HTTPS para garantizar la seguridad de los datos.
2. **Servidor de Aplicaciones:**
   * Gestiona la lógica del negocio, autenticación, y las interacciones con la base de datos.
   * Está alojado en Supabase, que proporciona servicios backend escalables y seguros.
3. **Base de Datos:**
   * Contiene toda la información crítica del sistema, como usuarios, profesionales, reservas y servicios.
   * Está alojada en Supabase, con acceso restringido para proteger la integridad de los datos.
4. **Servicios Externos:**
   * Incluye APIs externas como:
     + **OSRM API:** Para calcular rutas y navegación.
     + **Leaflet:** Para la visualización de mapas interactivos.
     + **Nominatim API:** Para transformar dirección ingresada a coordenadas y viceversa
5. **Conexiones:**
   * Todas las comunicaciones entre el cliente, el servidor y los servicios externos se realizan mediante protocolos seguros (HTTPS) para proteger los datos durante su transferencia.

**Relaciones de Despliegue**

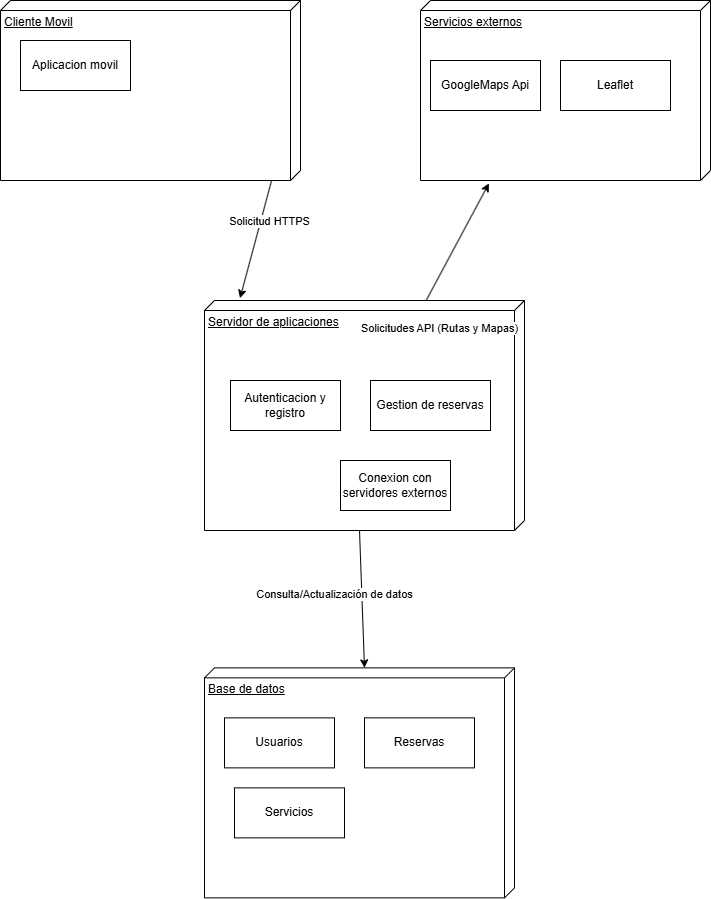
* El **cliente móvil** envía solicitudes al **servidor de aplicaciones** para realizar operaciones como registro, reserva y geolocalización.
* El **servidor de aplicaciones** interactúa con la **base de datos** para almacenar y recuperar información, y con los **servicios externos** para funcionalidades como mapas y rutas.
* Los **servicios externos** responden con los datos necesarios, como ubicaciones o cálculos de distancias.

**Diagrama de Despliegue**

El diagrama de despliegue muestra los nodos físicos y las conexiones entre ellos. En este caso, el sistema está compuesto por los siguientes nodos:

* **Cliente Móvil:** Aplicación móvil utilizada por los usuarios.
* **Servidor de Aplicaciones:** Backend alojado en Supabase.
* **Base de Datos:** Almacenamiento de datos en Supabase.
* **Servicios Externos:** APIs externas como OSRM, Geocoding y Leaflet.

**Figura 8:** Diagrama de despliegue

****

**Nota:** Muestra la distribución de los componentes del sistema en la infraestructura física, muestra la interacción de cliente móvil, servidor de aplicaciones y base de datos a través de conexiones seguras (HTTPS).

**Fuente:** Elaboración propia.

**Escenarios**

La **Escenarios** describe cómo los usuarios interactúan con el sistema para realizar las funcionalidades principales. Esta vista permite visualizar los escenarios clave desde la perspectiva de los actores involucrados y los procesos que ellos ejecutan en la plataforma.

**Actores Principales**

1. **Usuario Cliente:**
   * Representa a las personas que utilizan la aplicación para buscar servicios de peluquería, agendar reservas y gestionar sus perfiles.
2. **Usuario Profesional:**
   * Representa a los estilistas, barberos y peluqueros caninos que ofrecen sus servicios a través de la plataforma, gestionan horarios y reciben calificaciones de los clientes.
3. **Sistema:**
   * Representa la aplicación *EstilosExpress*, que centraliza las funcionalidades, conecta a los actores y asegura el flujo correcto de información entre ellos.

**Casos de Uso Principales**

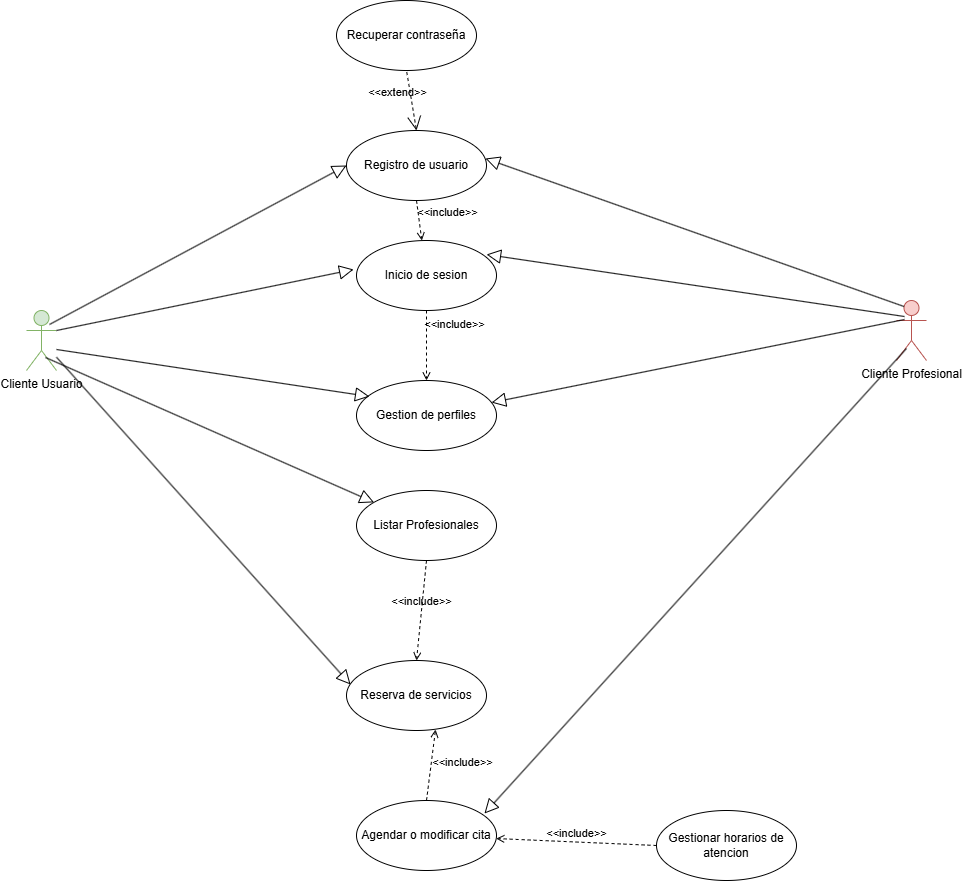
1. **Registro y Autenticación:**
   * Permite a los usuarios (clientes y profesionales) registrarse en la plataforma, iniciar sesión de manera segura y acceder a sus perfiles personalizados.
   * Este caso asegura la validación de datos y protege las credenciales de los usuarios mediante técnicas de encriptación.
2. **Geolocalización de Profesionales:**
   * Facilita la búsqueda de profesionales cercanos al cliente utilizando geolocalización en tiempo real.
   * Los usuarios pueden visualizar un mapa interactivo con la ubicación de los profesionales disponibles en un rango establecido.
3. **Reserva de Servicios:**
   * Permite a los clientes seleccionar a un profesional, elegir el servicio deseado, verificar horarios disponibles y confirmar una cita.
   * El sistema valida la disponibilidad del profesional y actualiza la información en la base de datos.
4. **Gestión de Perfiles:**
   * Proporciona a los usuarios la capacidad de editar sus datos personales, como nombre, foto y preferencias.
   * Para los profesionales, incluye la gestión de servicios ofrecidos, disponibilidad horaria y contacto con clientes.

**Relaciones entre Actores y Casos de Uso**

* **Usuario Cliente:** Interactúa directamente con los casos de uso para registrarse, buscar profesionales, realizar reservas y gestionar su perfil.
* **Usuario Profesional:** Utiliza las funcionalidades de la aplicación para registrarse, gestionar su disponibilidad, atender reservas y actualizar los servicios que ofrece.
* **Sistema:** Centraliza las operaciones, conectando a los actores y garantizando que todas las solicitudes se ejecuten de forma eficiente y segura.

Este análisis proporciona una perspectiva clara de cómo cada actor se relaciona con las funcionalidades principales, ayudando a garantizar que la experiencia de usuario sea fluida y alineada con los objetivos del sistema.

**Figura 9:** Relación actores y casos de uso.



**Nota:** Se muestra cómo los usuarios (Clientes y profesionales) interactúan con el sistema para realizar diversas funciones como registro o inicio de sesión.

**Fuente:** Elaboración propia.

## Requerimientos Técnicos

En esta sección se detallan todos los elementos técnicos requeridos para la implementación y desarrollo del proyecto *EstilosExpress*. Esta información incluye hardware, software, herramientas, servicios externos, elementos de seguridad, y tecnologías específicas empleadas para garantizar un sistema robusto, funcional y escalable.

**1. Dispositivos y Hardware**

Para llevar a cabo el desarrollo y pruebas de la aplicación *EstilosExpress*, se necesitan dispositivos que cumplan con las siguientes especificaciones técnicas mínimas:

**1.1 Computadoras para el Desarrollo**

* Estas son las estaciones de trabajo utilizadas por los desarrolladores para programar, probar y desplegar la aplicación.
* **Requisitos mínimos del hardware:**
  + **Procesador:** Intel i5 de 8ª generación o superior, AMD Ryzen 5 o equivalente.
  + **Memoria RAM:** 8 GB para un rendimiento fluido al compilar proyectos con Angular y ejecutar entornos virtuales.
  + **Almacenamiento:** SSD de 250 GB o más para almacenar herramientas de desarrollo y bases de datos locales.
  + **Sistema Operativo:** Windows 10, macOS Catalina o superior, o distribuciones de Linux (Ubuntu 20.04 recomendado).
  + **Pantalla:** Resolución mínima Full HD (1920x1080) para garantizar un entorno de trabajo cómodo.

**1.2 Dispositivos Móviles para Pruebas**

* **Propósito:** Validar el correcto funcionamiento de la aplicación en diferentes entornos y dispositivos móviles.
* **Requisitos mínimos:**
  + **Android:**
    - Sistema Operativo: Android 8.0 (Oreo) o superior.
    - Memoria RAM: 3 GB.
    - Resolución de pantalla: 720x1280 píxeles.
  + **iOS:**
    - Sistema Operativo: iOS 13 o superior.
    - Modelos soportados: iPhone SE (2020) o superior.
* **Dispositivos recomendados para pruebas:**
  + Gama media (Samsung Galaxy A52, Xiaomi Redmi Note 11).
  + Gama alta (iPhone 13, Samsung Galaxy S22).

**1.3 Conectividad y Redes**

* **Internet:** Conexión de banda ancha con una velocidad mínima de 50 Mbps para sincronizar repositorios, interactuar con APIs externas y realizar pruebas en la nube.
* **Ruteadores Wi-Fi:** Configuración estable para garantizar conexiones consistentes durante las pruebas de geolocalización.

**2. Software y Herramientas de Desarrollo**

El software utilizado en el proyecto incluye frameworks, lenguajes de programación, bibliotecas y plataformas diseñadas para optimizar el proceso de desarrollo y garantizar un alto rendimiento en dispositivos móviles.

**2.1 Frameworks y Librerías**

* **Angular con Ionic:** Framework utilizado para construir el frontend de la aplicación. Su capacidad híbrida permite compilar aplicaciones para Android e iOS a partir de una base de código única.
* **Leaflet:** Biblioteca de código abierto utilizada para la visualización y manejo de mapas interactivos.
* **Capacitor:** Plataforma que facilita la integración de funcionalidades nativas, como la geolocalización y la cámara, en dispositivos móviles.

**2.2 Lenguajes de Programación**

* **TypeScript:** Lenguaje principal del desarrollo, utilizado por Angular. Su tipado estático mejora la calidad del código y reduce errores.
* **HTML y CSS:** Lenguajes base para la construcción de interfaces de usuario intuitivas y personalizadas.
* **SQL:** Lenguaje empleado para consultas y gestión de datos en la base de datos PostgreSQL alojada en Supabase.

**2.3 Herramientas de Desarrollo**

* **Visual Studio Code:** Entorno de desarrollo integrado (IDE) elegido por su flexibilidad y extensiones que optimizan la experiencia del programador.
* **Git:** Sistema de control de versiones que permite al equipo trabajar colaborativamente, administrar ramas y realizar despliegues controlados.
* **Postman:** Herramienta utilizada para probar y depurar solicitudes API durante el desarrollo.

**3. Insumos y Servicios Externos**

El proyecto requiere servicios externos para extender la funcionalidad y mejorar la experiencia del usuario.

**3.1 Base de Datos**

* **Supabase:** Plataforma backend que gestiona la base de datos PostgreSQL del proyecto.
  + **Funcionalidades empleadas:**
    - Almacenamiento de usuarios, reservas y servicios.
    - Gestión de autenticación mediante tokens seguros.
    - APIs en tiempo real para sincronización de datos.

**3.2 APIs Externas**

* **OSRM API:**
  + Para la geolocalización y cálculo de rutas, facilitando a los usuarios encontrar profesionales cercanos.
* **Leaflet:**
  + Biblioteca empleada para representar visualmente los puntos en un mapa interactivo.
* **Nominatim API:** Para transformar dirección ingresada a coordenadas y viceversa

**3.3 Recursos Complementarios**

* **Certificados HTTPS:** Para garantizar conexiones seguras entre cliente y servidor.
* **Herramientas de análisis de rendimiento:** Para evaluar el comportamiento de la aplicación en tiempo de ejecución.

**4. Seguridad y Validaciones**

Para proteger la información del usuario y garantizar transacciones seguras, el sistema incorpora diversas medidas de seguridad:

**4.1 Hashing de Contraseñas**

* Se utiliza la librería **bcryptjs** para encriptar las contraseñas antes de almacenarlas en la base de datos, asegurando que nunca se guarden en texto plano.

**4.2 Autenticación Segura**

* A través de **Supabase**, los tokens de autenticación son generados y validados para cada sesión activa.

**4.3 Comunicación Segura**

* Todas las interacciones entre cliente y servidor se realizan bajo el protocolo **HTTPS**, protegiendo los datos en tránsito frente a ataques de intermediarios.

**4.4 Restricciones de Acceso**

* Datos sensibles entre usuarios (como correos electrónicos y ubicaciones exactas) están ocultos por defecto para evitar exposición innecesaria.

**5. Pruebas y Validación del Sistema**

Durante el desarrollo se realizaron diversas pruebas para garantizar la funcionalidad del sistema:

* **Pruebas Unitarias:** Implementadas con Jasmine y Karma para asegurar que cada componente funcione de manera individual.
* **Pruebas de Integración:** Validaciones para verificar que los módulos se comuniquen correctamente entre sí.
* **Pruebas de Compatibilidad:** Realizadas en dispositivos Android e iOS con distintas versiones para asegurar una experiencia homogénea.

## Desarrollo de la Solución

El desarrollo de **EstilosExpress** se enfocó en implementar una aplicación móvil moderna, eficiente y funcional que abordará las necesidades específicas de los usuarios y profesionales independientes en el ámbito de los servicios de peluquería. Este proceso implicó la planificación cuidadosa y ejecución técnica de diversas etapas clave, incluyendo la configuración de la base de datos, la implementación del backend, el desarrollo del frontend, y la integración de servicios externos para funcionalidades avanzadas. Cada etapa fue diseñada para garantizar una experiencia de usuario fluida, mientras se prioriza la seguridad y el rendimiento del sistema.

El enfoque adoptado durante el desarrollo se basó en prácticas ágiles que permitieron dividir el proyecto en entregables más pequeños y manejables, lo que facilitó la identificación y resolución temprana de posibles problemas técnicos. Se emplearon herramientas y tecnologías modernas como Supabase, Angular, Ionic y Leaflet, asegurando que la aplicación fuera escalable y compatible con dispositivos móviles actuales. Además, se realizaron pruebas exhaustivas en cada etapa del proceso para garantizar que el producto final cumpliera con los estándares de calidad requeridos, abordando tanto el rendimiento como la seguridad de los datos.

Esta metodología técnica rigurosa permitió no solo el desarrollo del producto mínimo viable (MVP), sino también sentar las bases para la evolución y mejora continua del sistema, proyectando futuras funcionalidades que enriquecerán aún más la experiencia tanto para los clientes como para los profesionales que utilicen la plataforma.

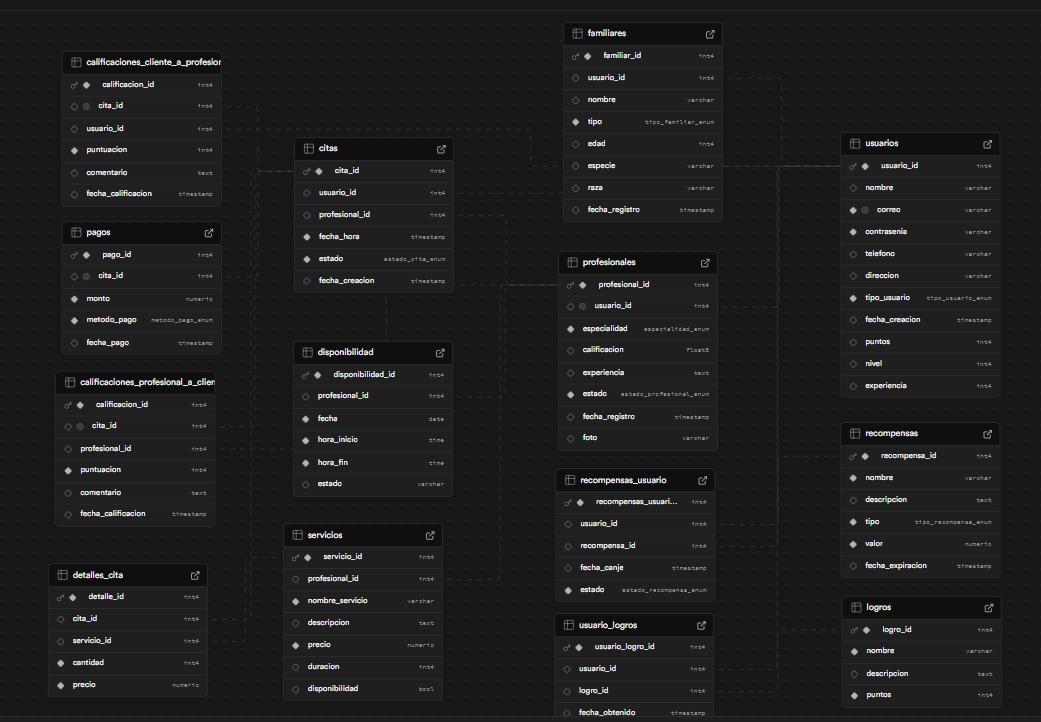
## 1. Configuración de la Base de Datos

La base de datos, gestionada con Supabase (PostgreSQL), es el corazón del sistema. Su diseño fue orientado a la integridad y la eficiencia, organizando los datos en tablas relacionadas para garantizar la consistencia de la información.

* Entidades principales:
  + Usuarios: Almacena información personal y credenciales de acceso.
  + Profesionales: Registra datos de los estilistas y barberos, incluyendo especialidades y disponibilidad.
  + Reservas: Gestor central de las citas entre clientes y profesionales.
  + Servicios: Lista los servicios ofrecidos, como cortes de cabello o baño para mascotas.
  + Calificaciones: Almacena reseñas y puntuaciones para medir la calidad de los servicios.

Seguridad: Las contraseñas se encriptan con bcryptjs para proteger la información sensible. Además, se aplicaron restricciones para evitar la duplicidad de correos electrónicos y datos erróneos.

**Figura 10:** Base de datos SupaBase



**Nota:** Muestra la estructura de la base de datos, detallando las entidades.

**Fuente:** Elaboración propia.

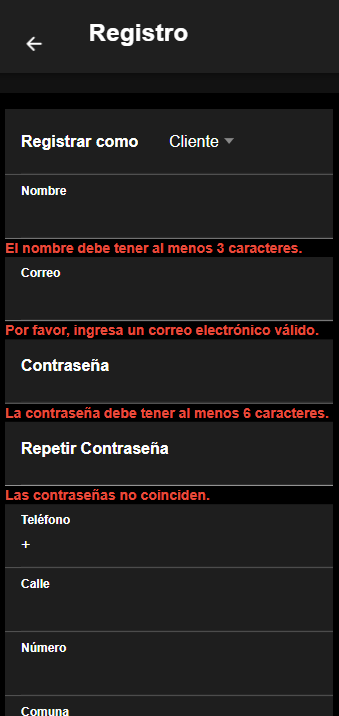
## 2. Implementación del Backend

El backend fue diseñado con **Supabase**, ofreciendo un sistema robusto y seguro para gestionar datos y lógica de negocio.

* **Funciones principales:**
  + Gestión de usuarios: Registro, inicio de sesión y recuperación de contraseñas.
  + Control de reservas: Creación, modificación y cancelación de citas.
  + Integración de servicios externos: Conexión con APIs como OSRM.
* **Validaciones:**
  + Verificación de correos electrónicos únicos.
  + Comprobación de disponibilidad de horarios antes de confirmar reservas.

**Pruebas:** Los endpoints fueron probados exhaustivamente con **Postman**, simulando diversas solicitudes y verificando respuestas correctas.

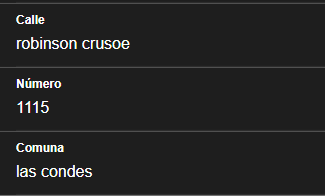
**Figura 11:** Registro de usuarios



**Nota:** Se muestran los campos necesarios para el registro de un usuario y las restricciones de cada campo.

**Fuente:** Elaboración propia.

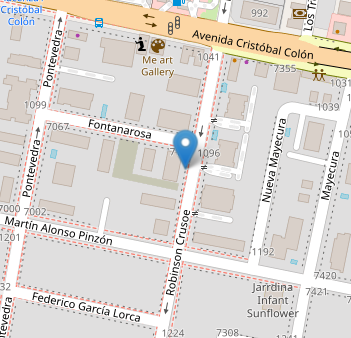
**Figura 12:** Registro de dirección



**Nota:** Campos necesarios para registrar una dirección.

**Fuente:** Elaboración propia.

**Figura 13:** Confirmación de dirección con mapa



**Nota:** Se muestra de manera interactiva el mapa para confirmar si la dirección que se ingresó es la correcta.

**Fuente:** Elaboración propia.

## 3. Configuración del Frontend

El frontend, desarrollado con **Angular** e **Ionic**, fue configurado para ofrecer una experiencia intuitiva y adaptativa en dispositivos móviles.

* **Pasos Realizados:**
  1. Configuración inicial del proyecto Angular con soporte para Ionic.
  2. Desarrollo de componentes clave como registro/login, mapa interactivo y gestión de reservas.
  3. Integración de servicios para conectarse con el backend utilizando APIs REST.
  4. Configuración de validaciones en tiempo real en formularios para asegurar datos consistentes.
  5. Implementación de mapas interactivos mediante **Leaflet**.

## 4. Integración de Servicios Externos

La integración de servicios externos fue esencial para extender las funcionalidades del sistema y ofrecer una experiencia más completa.

* **Pasos Realizados:**
  1. Configuración de OSRM API para obtener rutas y calcular distancias.
  2. Implementación de Leaflet para mostrar mapas interactivos y marcadores personalizados.
  3. implementación de Geocoding APIPara transformar dirección ingresada a coordenadas y viceversa
  4. Preparación para futuras integraciones con plataformas de pago y redes sociales.

## 5. Pruebas Técnicas

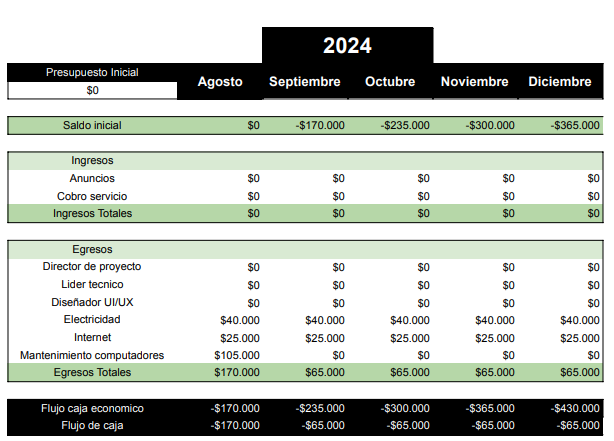
Las pruebas técnicas garantizaron la estabilidad y funcionalidad del sistema en diferentes escenarios.

* **Pasos Realizados:**
  1. **Pruebas Unitarias:** Evaluación de componentes individuales en el frontend utilizando herramientas como Jasmine.
  2. **Pruebas de Integración:** Validación de la comunicación entre el frontend y el backend, probando el flujo completo de operaciones.
  3. **Pruebas Funcionales:** Simulación de usuarios interactuando con el sistema, desde el registro hasta la confirmación de reservas.

# Factibilidad Económica

El análisis de la factibilidad económica del proyecto muestra que, debido a la falta de ingresos proyectados y presupuesto inicial, no es posible calcular la Tasa Interna de Retorno (TIR) para el proyecto. Además el cálculo de Valor Actual Neto (VAN) refleja que el proyecto tendrá un flujo de caja negativo hasta que la aplicación esté operativa en el mercado. Por otro lado, no se ha establecido un modelo de ingresos claro una vez que la aplicación esté operativa, lo que limita evaluar su sostenibilidad económica a largo plazo.

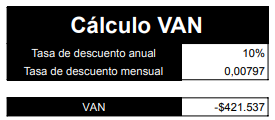
**Figura 14:** Flujo de caja



**Nota:** Se muestran los egresos económicos del proyecto hasta diciembre del año 2024. Dado que no se cuenta con un presupuesto inicial, ni ingresos previos antes del lanzamiento de la aplicación, se demuestra que el proyecto no tendrá un flujo de caja económico positivo hasta que la aplicación esté operativa.

**Fuente:** Elaboración propia.

**Figura 15:** Cálculo de van



**Nota:** El valor de van indica que el proyecto no genera valor económico, resultando en una pérdida.

**Fuente:** Elaboración propia.

# Conclusión

El proyecto EstiloExpress surgió como respuesta a la creciente demanda de servicios especializados en peluquería tanto para mascotas como para personas, enfrentando el problema de la falta de una herramienta eficiente para conectar clientes con profesionales cercanos. Para abordar esta necesidad, se ha desarrollado una aplicación móvil diseñada para facilitar la búsqueda de servicios, permitiendo a los usuarios encontrar profesionales basados en su ubicación y disponibilidad, además, de agendar citas de manera rápida y sencilla.

Desde la concepción de la idea, el proceso comenzó con la identificación del problema y el diseño de una solución centrada en el usuario. Posteriormente, llevamos a cabo una investigación tecnológica para definir las herramientas más adecuadas para el desarrollo del Producto Mínimo Viable (MVP). Se opto por el framework ionic con angular para construir una aplicación híbrida que funciones tanto para Android como en IOS, logrando una mayor eficiencia en tiempos y costos de desarrollo. La base de datos elegida fue SupaBase, debido a su capacidad de trabajar con SQL, lo que facilitó operaciones clave como inserción, modificación y consultas de datos.

Para el desarrollo, se utilizó Visual Studio Code, un entorno de desarrollo gratuito que se adaptó perfectamente a las necesidades del proyecto, al proporcionar herramientas intuitivas para codificación y depuración. Las pruebas finales realizadas garantizaron que la aplicación cumpliera con los requisitos funcionales y ofreciera una experiencia de usuario fluida.

En conclusión, EstiloExpress no solo resuelve un problema real en el mercado, sino que, también establece una base sólida para futuras mejoras y expansión.

# Mejoras a futuro

* **Pagos en línea:** implementación de opciones de pago en línea seguras y rápidas. Por ejemplo, la implementación de pagos a través de plataformas como WebPay o Paypal, permitiendo a los usuarios realizar pagos con tarjeta de débito o crédito.
* **Sistema de reseñas y calificaciones:** Se añadirá un sistema donde los usuarios pueden dejar reseñas y calificaciones sobre los profesionales. Después de cada servicio el usuario podrá calificar al profesional con puntuación de 1 a 5 y dejar comentarios sobre su experiencia, ayudando a otros usuarios a elegir el mejor servicio.
* **Optimización del rendimiento:** Mejora en la velocidad de la aplicación para asegurar tiempos de carga rápidos, así mejorando la experiencia de usuario.
* **Integración con aplicaciones de viaje:** Mejorar la geolocalización que utilizamos para que al momento de que un usuario reserva un servicio, el profesional con la ayuda de Google Maps o Waze calcule la mejor ruta para llegar al domicilio del usuario, además se puede integrar Uber o Cabify para opciones con transporte público.

# Acrónimos

* MVP**:** Producto Mínimo Viable.
* UI**:** Interfaz de Usuario.
* UX**:** Experiencia de Usuario.
* API**:** Interfaz de Programación de Aplicaciones.
* CRUD**:** Crear, Leer, Actualizar y Eliminar.
* HTTPS**:** Protocolo Seguro de Transferencia de Hipertexto.
* JWT: Token Web JSON.
* VAN: Valor Actual Neto.
* TIR: Tasa Interna de Retorno.
* SQL: Lenguaje De Consulta Estructurado.

# Bibliografía

* **Objetivos generales y específicos: qué son y cómo redactarlos.**Publicado el 16 de agosto de 2024. Consultado el 10 de octubre de 2024.  
  <https://asana.com/es/resources/general-and-specific-objetives>
* **Información sobre las fases y estructura del proyecto.**Publicado el 25 de enero de 2021. Consultado el 5 de septiembre de 2024.  
  <https://www.atlassian.com/es/work-management/project-management/phases>
* **Descripción y el propósito del proyecto.**Publicado el 9 de abril de 2022. Consultado el 10 de octubre de 2024.  
  <https://es.smartsheet.com/content/project-description>
* **Cómo describir un problema para que todos entiendan.**Publicado el 10 de septiembre de 2024. Consultado el 11 de septiembre de 2024.  
  <https://blog.softexpert.com/es/como-describir-problema/>
* **Cómo funciona la geolocalización móvil.**Publicado el 30 de mayo de 2024. Consultado el 10 de octubre de 2024.  
  <https://www.reanimasoluciones.com/actualidad/278-como-funciona-la-geolocalizacion-movil#geolocalizacion-movil-como-funciona>
* **Sistema de reservas con gestión de horarios.**Publicado el 20 de febrero de 2023. Consultado el 10 de agosto de 2024.  
  <https://www.siteminder.com/es/motor-de-reservas-en-linea>
* **Desarrollar un sistema de soporte al usuario dentro de la aplicación.**Actualizado el 24 de abril de 2024 (publicado el 7 de enero de 2020). Consultado el 11 de septiembre de 2024.  
  <https://blog.invgate.com/es/software-soporte-tecnico>
* **Tecnologías utilizadas.**Publicado el 11 de mayo de 2022. Consultado el 22 de noviembre de 2024.  
  <https://www.armadilloamarillo.com/blog/mejores-tecnologias-para-desarrollo-de-aplicaciones-moviles/>
* **Geolocalización API.**Publicado el 10 de enero de 2024. Consultado el 2 de noviembre de 2024.  
  <https://jesusyesares.com/visualizacion-datos/que-es-leaflet-libreria-javascript/>