

1. Introducción y Resumen Ejecutivo

YH Pure Zone es una empresa especializada en servicios de limpieza profesional a domicilio. Su modelo operativo actual se basa en la gestión manual de clientes y citas a través de canales de comunicación directos (llamadas, WhatsApp), lo cual representa un cuello de botella para el crecimiento y la eficiencia.

Este proyecto define la creación de una **plataforma web integral** diseñada desde su origen para ser funcional e interactiva. El objetivo no es simplemente mostrar información, sino construir una herramienta digital que permita a los clientes **autogestionar todo el ciclo de servicio**: desde obtener una cotización instantánea y agendar en tiempo real, hasta consultar su historial. Para el negocio, la plataforma centralizará y automatizará las operaciones, transformando la manera en que se gestionan los servicios.

2. Justificación del Proyecto

La transición de un modelo operativo manual a una plataforma digital automatizada es una inversión estratégica fundamental para el futuro de YH Pure Zone. La justificación se sustenta en:

- **Ineficiencia del Modelo Actual:** La gestión manual es intensiva en tiempo, propensa a errores humanos (ej. doble agendamiento, pérdida de información del cliente) y no es escalable. Cada nuevo cliente añade una carga administrativa linealmente creciente.
- **Expectativas del Cliente Moderno:** El consumidor actual valora la inmediatez y la autonomía. La necesidad de esperar una respuesta para una cotización o para confirmar una disponibilidad es un punto de fricción que causa la pérdida de clientes, quienes optarán por alternativas más rápidas y convenientes.
- **Creación de un Activo de Datos:** Una plataforma funcional centraliza toda la información en una base de datos. Cada cita, cliente y servicio registrado se convierte en un dato valioso que permite analizar tendencias (ej. ¿cuál es el servicio más rentable?), identificar clientes recurrentes y tomar decisiones estratégicas informadas.
- **Ventaja Competitiva Sostenible:** La implementación de una herramienta de autogestión posiciona a YH Pure Zone como un líder en innovación y servicio al cliente en el mercado local. La barrera tecnológica crea una ventaja competitiva difícil de replicar para competidores que operan de manera tradicional.

3. Objetivos del Proyecto

Objetivo General:

Desarrollar una plataforma web integral y robusta que automatice por completo el ciclo de vida del cliente para YH Pure Zone, permitiendo la cotización, el agendamiento y la gestión de servicios de forma autónoma por parte del usuario, mientras se optimizan y centralizan las operaciones internas del negocio.

Objetivos Específicos:

1. **Implementar un sistema de agendamiento en tiempo real** que se conecte a una base de datos para mostrar horarios disponibles y permita a los clientes reservar servicios 24/7 sin intervención humana.
2. **Desarrollar una calculadora de cotizaciones interactiva** que genere precios precisos al instante, basados en las selecciones y personalizaciones del usuario.
3. **Construir un portal de cliente seguro (con login)** donde los usuarios puedan consultar su historial de servicios, ver sus próximas citas y gestionar su información personal.
4. **Centralizar el 100% de la información operativa** (clientes, citas, servicios, precios, reseñas) en una base de datos relacional (PostgreSQL/MySQL) como única fuente de verdad.
5. **Crear un panel de administración interno** que permita al personal de YH Pure Zone gestionar la disponibilidad, los servicios, los precios, las citas y los clientes de manera centralizada.
6. **Automatizar las comunicaciones clave con el cliente**, como las confirmaciones de reserva y los recordatorios de citas, a través de notificaciones por correo electrónico.
7. **Reducir la dependencia de la gestión manual vía WhatsApp/teléfono en un 90%** para tareas de agendamiento y cotización en los primeros seis meses de operación.

4. Temario / Módulos Funcionales de la Plataforma

La plataforma se concibe como un sistema modular, donde cada componente cumple una función específica dentro del ecosistema digital.

- **Módulo 1: Núcleo de la Aplicación y Base de Datos (Backend)** ○ Diseño y creación del esquema de la base de datos.
 - Desarrollo de una API RESTful para servir como puente entre la base de datos y la interfaz de usuario.
 - Lógica de negocio para la gestión de disponibilidad, precios y reglas de agendamiento.
- **Módulo 2: Sistema de Agendamiento (Frontend - Público)**
 - Interfaz de calendario interactivo que consume datos de la API para mostrar fechas/horas disponibles.
 - Flujo de reserva multi-paso: 1) Selección de servicio, 2) Selección de fecha/hora, 3) Ingreso de datos, 4) Confirmación.
- **Módulo 3: Calculadora de Cotizaciones (Frontend - Público)**
 - Componente interactivo que permite al usuario seleccionar y personalizar servicios.

- Cálculo de precios en tiempo real.
- Integración directa con el Módulo de Agendamiento para reservar la cotización generada.
- **Módulo 4: Portal de Cliente (Frontend - Privado)**
 - Sistema de autenticación (registro e inicio de sesión).
 - Panel de usuario para visualizar historial de citas y citas futuras.
 - Funcionalidades para la gestión de su perfil.
- **Módulo 5: Panel de Administración (Frontend - Privado)**
 - Dashboard con vista general de las citas del día/semana/mes.
 - Herramientas CRUD (Crear, Leer, Actualizar, Borrar) para gestionar Servicios, Clientes y Citas.
 - Control sobre la disponibilidad (ej. bloquear días festivos).

5. Levantamiento de Requerimientos Funcionales y No Funcionales : Plataforma Interactiva YH Pure Zone

Nuestro propósito es definir de manera inequívoca las capacidades y funciones que el sistema debe proveer a sus distintos usuarios. Cada requerimiento describe una acción, comportamiento o pieza de información que el sistema debe gestionar, sirviendo como guía fundamental para el equipo de desarrollo y como base para las pruebas de validación.

- Perfiles de Usuario (Actores)

Se han identificado tres perfiles de usuario principales que interactuarán con el sistema:

- **Visitante (No autenticado):** Cualquier usuario que accede a la plataforma sin haber iniciado sesión. Su objetivo es explorar servicios, obtener cotizaciones y, potencialmente, convertirse en cliente.
- **Cliente (Autenticado):** Un usuario que ha creado una cuenta y ha iniciado sesión. Su objetivo es gestionar sus citas, consultar su historial y administrar su perfil.
- **Administrador (Personal de YH Pure Zone):** Un usuario con privilegios especiales que accede a un panel interno para gestionar la totalidad de las operaciones del negocio a través de la plataforma.

- Requerimientos Funcionales por Módulo

A continuación, se especifican los requerimientos funcionales, agrupados por los módulos definidos en el alcance del proyecto.

- Módulo 1: Gestión de Cuentas y Autenticación

ID	REQUERIMIENTO	ACTOR(ES)	DESCRIPCIÓN
----	---------------	-----------	-------------

RF-001	Registro de cuenta	Visitante	El sistema debe permitir a un visitante crear una nueva cuenta de cliente proporcionando nombre, correo electrónico, número de teléfono y una contraseña.
RF-002	Inicio de sesión	Visitante, Cliente	El sistema debe permitir a los usuarios (Clientes y Administradores) iniciar sesión utilizando su correo electrónico y contraseña.
RF-003	Cierre de sesión	Cliente, Administrador	El sistema debe permitir a un usuario autenticado cerrar su sesión de forma segura.
RF-004	Recuperación de contraseña	Visitante, Cliente	El sistema debe proporcionar una funcionalidad para que un usuario pueda restablecer su contraseña a través de su correo electrónico registrado.
RF-005	Acceso diferenciado	Administrador	El sistema debe redirigir a los usuarios al panel correspondiente (Portal de Cliente o Panel de Administración) según su rol tras el inicio de sesión.

- Módulo 2: Calculadora de Cotizaciones

ID	REQUERIMIENTO	ACTOR(ES)	DESCRIPCIÓN
RF-006	Visualización de servicios	Visitante, Cliente	El sistema debe mostrar las categorías principales de servicios (ej. Muebles, Vehículos) de forma clara.
RF-007	Personalización de servicios	Visitante, Cliente	El sistema debe permitir al usuario seleccionar opciones específicas para un servicio (ej. # de puestos de un sofá, tipo de vehículo) para personalizar la cotización.
RF-008	Cálculo de precio en tiempo real	Visitante, Cliente	El sistema debe calcular y mostrar el costo total del servicio instantáneamente a medida que el usuario modifica las opciones.
RF-009	Continuidad a la reserva	Visitante, Cliente	El sistema debe permitir al usuario proceder directamente al agendamiento con los servicios y el precio ya calculados en la cotización.

- Módulo 3: Sistema de Agendamiento

ID	REQUERIMIENTO	ACTOR(ES)	DESCRIPCIÓN
RF-010	Visualización de disponibilidad	Visitante, Cliente	El sistema debe mostrar un calendario con las fechas y los bloques horarios disponibles para el servicio seleccionado.
RF-011	Restricción de horarios	Visitante, Cliente	El sistema no debe permitir la selección de fechas pasadas, ni de fechas/horas que ya están completamente reservadas o bloqueadas por el administrador.

RF-012	Formulario de reserva	Visitante, Cliente	El sistema debe presentar un formulario para que el usuario ingrese o confirme sus datos personales y la dirección donde se realizará el servicio.
RF-013	Confirmación de cita	Visitante, Cliente	Tras completar la reserva, el sistema debe mostrar una página de confirmación exitosa y registrar la cita en la base de datos con estado "Confirmada".
RF-014	Notificación de nueva cita	Cliente, Administrador	El sistema debe enviar automáticamente un correo electrónico de confirmación al cliente y una notificación de nueva cita al administrador.

- Módulo 4: Portal de Cliente

ID	REQUERIMIENTO	ACTOR(ES)	DESCRIPCIÓN
RF-015	Visualización de citas futuras	Cliente	El sistema debe mostrar al cliente una lista de todas sus citas agendadas y pendientes, con detalles como servicio, fecha, hora y costo.
RF-016	Visualización de historial de citas	Cliente	El sistema debe mostrar al cliente un historial de todos los servicios que ha completado anteriormente.
RF-017	Cancelación de cita	Cliente	El sistema debe permitir al cliente cancelar una cita futura, sujeto a las reglas de negocio (ej. con más de 24 horas de antelación).
RF-018	Gestión de perfil	Cliente	El sistema debe permitir al cliente ver y modificar su información personal (nombre, teléfono, dirección).

- Módulo 5: Panel de Administration

ID	REQUERIMIENTO	ACTOR(ES)	DESCRIPCIÓN
RF-019	Dashboard de citas	Administrador	El sistema debe presentar un panel principal con una vista general de las citas del día, la semana y el mes.
RF-020	Gestión de citas	Administrador	El sistema debe permitir al administrador ver, filtrar, y cambiar el estado de todas las citas (ej. de "Confirmada" a "Completada").
RF-021	Creación manual de citas	Administrador	El sistema debe permitir al administrador crear una nueva cita en nombre de un cliente (para reservas hechas por teléfono).
RF-022	Gestión de servicios y precios	Administrador	El sistema debe proporcionar una interfaz para crear, editar y eliminar servicios, así como para definir sus opciones y estructura de precios.
RF-023	Gestión de clientes	Administrador	El sistema debe permitir al administrador ver la lista de todos los clientes registrados y consultar su historial de servicios.

RF-024	Gestión de disponibilidad	Administrador	El sistema debe permitir al administrador bloquear fechas o rangos horarios específicos (ej. vacaciones, festivos) para que no puedan ser reservados por los clientes.
---------------	---------------------------	---------------	--

- Requerimientos No Funcionales por Modulo

Estos requerimientos establecen los estándares de calidad, rendimiento, seguridad y operatividad que el sistema debe cumplir para garantizar una experiencia de usuario satisfactoria y una operación de negocio confiable.

- **Usabilidad**

Los requerimientos de usabilidad aseguran que la plataforma sea fácil de usar, intuitiva y accesible para todos los perfiles de usuario.

ID	REQUERIMIENTO	DESCRIPCIÓN
RNF001	Intuitividad en el flujo principal:	Un usuario nuevo debe poder completar el proceso de cotización y agendamiento de un servicio en menos de 3 minutos sin necesidad de un tutorial o ayuda externa.
RNF002	Diseño adaptable (Responsive):	La interfaz de la plataforma debe ser 100% funcional y visualmente coherente en los principales tamaños de pantalla, incluyendo computadores de escritorio, tabletas y teléfonos móviles.
RNF003	Consistencia de la interfaz:	Todos los elementos de la interfaz (botones, formularios, menús) deben seguir un diseño consistente a lo largo de toda la aplicación para evitar confusiones.
RNF004	Retroalimentación al usuario:	El sistema debe proporcionar mensajes claros e inmediatos para confirmar acciones (ej. "Cita agendada con éxito") o para notificar errores (ej. "La contraseña es incorrecta"). Todo el texto debe estar en español colombiano.

- **Rendimiento**

Estos requerimientos definen la velocidad y capacidad de respuesta de la plataforma bajo diferentes condiciones de carga.

ID	REQUERIMIENTO	DESCRIPCIÓN
RNF005	Tiempo de carga inicial:	El tiempo de carga completo de la página principal no debe exceder los 3 segundos en una conexión de internet móvil 4G estándar.
RNF006	Tiempo de respuesta de la API:	Las interacciones clave que dependen del servidor (ej. verificar disponibilidad en el calendario, calcular cotización) deben tener un tiempo de respuesta inferior a 500 milisegundos.

RNF007	Concurrencia de usuarios:	El sistema debe ser capaz de soportar a 50 usuarios simultáneos realizando transacciones sin una degradación perceptible en el rendimiento.
---------------	---------------------------	---

- **Seguridad**

La seguridad es crítica para proteger los datos de los clientes y la integridad del negocio.

ID	REQUERIMIENTO	DESCRIPCIÓN
RNF008	Almacenamiento de contraseñas:	Todas las contraseñas de los usuarios deben ser almacenadas en la base de datos utilizando un algoritmo de hash fuerte e irreversible (ej. Bcrypt). Nunca deben guardarse en texto plano.
RNF009	Comunicación encriptada:	Toda la comunicación entre el navegador del cliente y el servidor debe estar cifrada mediante el protocolo HTTPS (SSL/TLS).
RNF010	Protección contra vulnerabilidades:	La plataforma debe estar protegida contra las vulnerabilidades web más comunes del Top 10 de OWASP, incluyendo Inyección SQL y Cross-Site Scripting (XSS).
RNF011	Gestión de sesiones:	Las sesiones de usuario deben expirar automáticamente después de un período de inactividad (ej. 60 minutos) para prevenir accesos no autorizados en dispositivos compartidos.

- **Disponibilidad y Confiabilidad**

Estos requerimientos aseguran que la plataforma esté operativa y funcione correctamente la mayor parte del tiempo.

ID	REQUERIMIENTO	DESCRIPCIÓN
RNF012	Tiempo de actividad (Uptime):	La plataforma debe tener una disponibilidad garantizada del 99.5%, excluyendo ventanas de mantenimiento planificadas y notificadas.
RNF013	Manejo de errores:	El sistema debe ser robusto y manejar los errores de forma controlada (ej. un fallo en el envío de un correo no debe impedir que se complete una reserva), mostrando un mensaje amigable al usuario sin colapsar.
RNF014	Copias de seguridad:	Se deben realizar copias de seguridad automáticas y completas de la base de datos al menos una vez cada 24 horas, con una política de retención de 7 días.

- **Mantenibilidad y Escalabilidad**

Estos requerimientos facilitan la evolución, corrección y crecimiento futuro de la plataforma.

ID	REQUERIMIENTO	DESCRIPCIÓN
RNF015	Código documentado y legible:	El código fuente debe seguir estándares de codificación consistentes y estar debidamente comentado para facilitar el mantenimiento y la incorporación de nuevos desarrolladores.

RNF016	Arquitectura modular:	El sistema debe ser diseñado con una arquitectura modular, permitiendo que nuevas funcionalidades (ej. integración de pasarelas de pago) se puedan añadir en el futuro con un impacto mínimo en el código existente.
RNF017	Escalabilidad de la infraestructura:	La infraestructura del servidor debe estar preparada para escalar horizontalmente (añadir más servidores) para soportar un futuro aumento en el tráfico de usuarios.

- **Compatibilidad**

Estos requerimientos definen los entornos en los que la plataforma debe funcionar correctamente.

ID	REQUERIMIENTO	DESCRIPCIÓN
RNF018	Compatibilidad de navegadores:	La plataforma debe ser totalmente compatible con las dos últimas versiones estables de los principales navegadores web: Google Chrome, Mozilla Firefox y Safari.
RNF019	Compatibilidad de sistemas operativos:	La experiencia del usuario debe ser consistente y funcional en los principales sistemas operativos móviles (iOS y Android) y de escritorio (Windows y macOS).

6. Esquema de Base de Datos

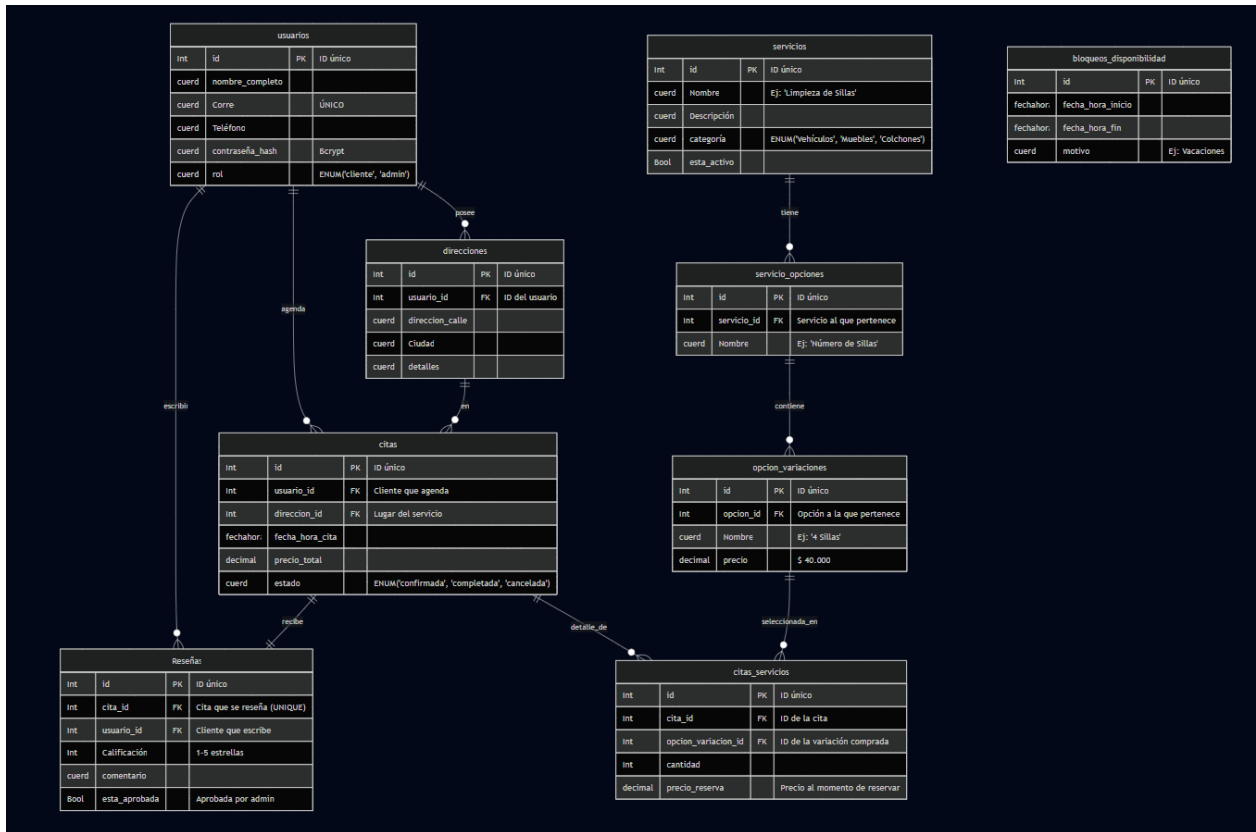
La base de datos se diseñó bajo un modelo relacional (SQL), utilizando MySQL. La estructura se centra en la normalización para garantizar la integridad de los datos.

Especificación de Datos

- **Tabla usuarios:** Almacena los datos de acceso y el rol (cliente o admin) de todas las personas que usan la plataforma. Es la tabla central de autenticación.
- **Tabla direcciones:** Guarda las diferentes ubicaciones de un usuario. Un cliente puede tener múltiples direcciones.
- **Tabla servicios:** Es el catálogo genérico. Define la categoría (ej. "Muebles") y el nombre (ej. "Limpieza de Sillas de Comedor"), pero no contiene precios.
- **Tabla servicio_opciones:** Tabla intermedia que define las *categorías de personalización* de un servicio. (Ej: El servicio "Limpieza de Sillas" tiene las opciones "Número de Sillas" y "Tipo de Limpieza").
- **Tabla opcion_variaciones:** Aquí es donde residen los precios. Define las *opciones específicas* y su costo. (Ej: La opción "Número de Sillas" tiene las variaciones "4 Sillas" (\$0) y "6 Sillas" (\$10.000)).
- **Tabla citas:** El registro de cada reserva. Conecta un usuario_id con una direccion_id en una fecha_hora_cita, guardando el precio_total y su estado (confirmada, completada, cancelada).
- **Tabla citas_servicios:** Tabla intermedia que conecta una Cita con las Variaciones específicas que el cliente compró. Guarda el cita_id y el opcion_variacion_id, siendo esta la clave de todo el flujo de agendamiento.

- **Tabla reseñas:** Contiene los comentarios y calificaciones (1-5) que los clientes dejan después de un servicio completado, vinculadas a una cita_id para verificar su autenticidad.
- **Tabla bloqueos_disponibilidad:** Permite al administrador bloquear fechas u horas, haciendo que no aparezcan en el calendario de la calculadora.

Modelo Entidad Relación



7.Arquitectura de la Aplicación

- **Frontend (La Parte Visual)**

Es lo que el usuario ve y con lo que interactúa en su navegador.

- **Tecnologías:** Usaremos **HTML** para la estructura, **CSS** para el estilo y **JavaScript** para la interactividad.
- **Framework:** Utilizaremos **Bootstrap 5** para construir un diseño moderno y adaptable a dispositivos móviles (responsive) de manera rápida, aprovechando sus componentes predefinidos.

-
- **Backend (El Cerebro)**

Es el servidor que se ejecuta de forma remota y contiene toda la lógica del negocio.

- **Tecnología:** Usaremos **Node.js** como entorno de ejecución.
- **Framework:** Implementaremos el servidor y la API RESTful con **Express.js** para gestionar las rutas y las peticiones de forma ordenada.

- **Base de Datos (La Memoria)**

Es donde se almacenará toda la información de forma persistente y segura.

- **Tecnología:** Usaremos **MySQL** como nuestro sistema gestor de base de datos relacional para guardar todas las tablas (usuarios, citas, servicios, etc.).

- **Flujo de Trabajo**

El proceso será el siguiente: El **Frontend** (hecho con HTML, CSS, JS y Bootstrap) enviará peticiones a la **API** creada con **Express**. El servidor en **Node.js** procesará estas peticiones, interactuará con la base de datos **MySQL** para leer o guardar información y devolverá una respuesta al Frontend para que este actualice la vista del usuario.

8. Enfoque Metodológico: Modelo Híbrido (Cascada + Ágil)

Para la gestión y el desarrollo de la plataforma YH Pure Zone, se optó por un modelo de desarrollo híbrido que combina la robustez de la metodología en Cascada con la flexibilidad de un enfoque Ágil (Iterativo).

Este enfoque nos permitió construir una base de backend sólida antes de iterar rápidamente en la experiencia de usuario del frontend.

Proceso de Implementación

El desarrollo se dividió en dos fases principales:

Fase 1: Desarrollo del Backend (Enfoque Cascada) Esta fase siguió un orden secuencial estricto para construir el "cerebro" de la aplicación:

1. **Diseño de la Base de Datos:** Se diseñó y creó el script SQL para la base de datos en MySQL.
2. **Implementación de API:** Se desarrollaron todos los controladores y rutas de la API (para autenticación, gestión de citas, clientes, etc.)
3. **Pruebas Unitarias de API:** Cada *endpoint* de la API fue probado de forma aislada (usando Postman) para asegurar su correcto funcionamiento antes de construir cualquier interfaz visual.

Fase 2: Desarrollo del Frontend (Enfoque Ágil e Iterativo) Una vez que el backend estuvo validado, la construcción de la interfaz de usuario se manejó de forma iterativa y modular:

- **Construcción por Módulos:** Se trabajó en módulos funcionales completos (Calculadora, Panel de Admin, Panel de Cliente).
- **Detección y Corrección Temprana:** Este enfoque permitió descubrir problemas críticos que el diseño inicial no previó. El ejemplo más importante fue la corrección de la lógica de precios (servicio_id vs. opcion_variacion_id), que requirió refactorizar la base de datos, el backend y el frontend simultáneamente.
- **Mejoras Incrementales:** Se añadieron nuevas APIs (como GET /api/cliente/citas o POST /api/cliente/resenas) a medida que el frontend las necesitaba, en lugar de definir las todas al inicio.

Ventajas de este Enfoque Híbrido:

- **Solidez:** Se aseguró que la lógica de negocio y la base de datos fueran estables antes de construir la interfaz.
- **Flexibilidad:** Permite adaptar y mejorar la experiencia del usuario sobre la marcha, respondiendo a problemas (como los errores 404 o 500) y añadiendo funcionalidades no contempladas (como las notificaciones de reseñas).

Ventajas para el Proyecto

La adopción de Scrum ofrecerá las siguientes ventajas estratégicas:

- **Gestión de la Complejidad:** Divide el proyecto en partes manejables, facilitando el desarrollo y el seguimiento.
- **Visibilidad del Progreso:** Proporciona resultados tangibles y funcionales al final de cada Sprint, permitiendo una evaluación continua.
- **Adaptabilidad:** Permite incorporar cambios o nuevas ideas de forma ordenada entre Sprints, sin perturbar el desarrollo en curso.
- **Detección Temprana de Riesgos:** Al construir y probar en ciclos cortos, los problemas se identifican y resuelven de manera temprana, reduciendo el riesgo de fallos mayores en etapas avanzadas.