



I.E.S GRAN CAPITÁN

GRADO FORMATIVO DE GRADO SUPERIOR
EN APLICACIONES MULTIPLATAFORMA.

PROYECTO INTEGRADO

Manual Técnico

Autores:

- Antonio Jesús Martínez Díaz – 2º DAM
- Rocío Zuara Jiménez – 2º DAM
- Lorena Barea Rot – 2º DAM

INDICE:

1.Introducción

2.Objetivos

3.Estudio previo

- 3.1 Estado actual
- 3.2 Estudio de soluciones existentes

4.Diseño Backed

- 4.1 Diseño general
- 4.2 Diseño detallado

5.Diseño Frontend

6.Implantación

7.Conclusiones

8.Bibliografía

1. INTRODUCCION:

Lavadero Sepúlveda es un sistema integral y multiplataforma diseñado para la gestión completa de citas en lavaderos de vehículos. El proyecto combina aplicaciones de escritorio, web y móvil, permitiendo digitalizar por completo los procesos de reserva, administración y atención al cliente. Su enfoque modular garantiza una experiencia fluida tanto para los usuarios como para el personal del negocio, adaptándose a las necesidades actuales de eficiencia, accesibilidad y modernización.

2. OBJETIVOS

Objetivos Generales

Desarrollar una plataforma flexible, moderna y centralizada que permita gestionar de forma eficiente las citas, clientes, servicios y recursos de un lavadero de vehículos, optimizando el flujo de trabajo, reduciendo tareas manuales y ofreciendo una experiencia de uso ágil e intuitiva.

Objetivos Específicos

1. Digitalización completa del proceso de reservas

El sistema permitirá a los clientes reservar, modificar o cancelar citas de manera online desde cualquier dispositivo. Esto eliminará la gestión manual, reducirá errores, agilizará los tiempos de atención y mejorará la organización diaria del personal.

Incluye funcionalidades como:

- Control de disponibilidad en tiempo real.
- Registro automatizado de citas.
- Gestión de servicios y horarios.

2. Desarrollo de un sistema multiplataforma

La plataforma ofrecerá varios puntos de acceso para garantizar la máxima accesibilidad:

Aplicación web para clientes y administradores.

Aplicación móvil (Android e iOS) orientada a clientes y personal.

Aplicación de escritorio para la gestión operativa interna.

Todas las interfaces estarán sincronizadas mediante una arquitectura centralizada.

3. Centralización y sincronización de la información

Toda la información del negocio —clientes, empleados, citas, servicios, historial y métricas— se almacenará en una base de datos centralizada, asegurando coherencia y disponibilidad en tiempo real.

Esto permitirá:

- Acceso unificado a los datos.
- Informes automáticos, estadísticas y seguimiento operativo.
- Evitar duplicidades o inconsistencias.

4. Mejora de la eficiencia operativa y de la experiencia del cliente

La plataforma incorporará herramientas que agilizan la interacción con el cliente y la organización interna:

- Notificaciones automáticas por correo o SMS.
- Recordatorios de citas y avisos de cambios.
- Gestión optimizada de turnos y cargas de trabajo.
- Reducción de tiempos muertos y colas.

5. Flexibilidad, modularidad y escalabilidad

El sistema se diseñará para permitir la incorporación de nuevas funcionalidades sin afectar su funcionamiento.

Entre las posibles extensiones futuras se incluyen:

- Integración con pasarelas de pago.
- Control de inventario.
- Programas de fidelización.
- Expansión a franquicias o múltiples sedes.

6. Seguridad y protección de datos

Se implementarán medidas de seguridad que garanticen la protección de la información y el cumplimiento del RGPD:

- Autenticación segura.
- Cifrado de contraseñas.
- Conexiones HTTPS.
- Gestión segura de roles y permisos.

7. Durabilidad, mantenibilidad y buenas prácticas

El proyecto se desarrollará con tecnologías estándar y ampliamente soportadas, asegurando:

- Código limpio y documentado.
- Arquitectura modular para facilitar actualizaciones y correcciones.
- Mantenimiento eficiente sin interrumpir la actividad diaria del negocio.

3. ESTUDIO PREVIO

3.1. Estado actual

En la actualidad, la mayoría de los lavaderos de vehículos continúan gestionando sus citas de forma manual, a través de agendas físicas, notas improvisadas o mediante atención telefónica directa. Este modelo, aunque tradicional, presenta importantes limitaciones que afectan tanto al funcionamiento interno como a la calidad del servicio ofrecido al cliente.

Entre los principales problemas detectados se encuentran:

- **Confusión de horarios y solapamiento de reservas:**

La ausencia de un sistema automatizado provoca que, en momentos de alta demanda, las citas puedan anotarse de forma incorrecta o duplicada. Esto genera descoordinación, retrasos y una percepción negativa por parte del cliente.

- **Tiempos elevados en la atención y registro de citas:**

El personal debe invertir tiempo en atender llamadas, confirmar disponibilidad, buscar huecos libres y registrar manualmente cada cita. Este proceso no solo es lento, sino que aumenta el riesgo de errores humanos.

- **Falta de control y seguimiento del historial de clientes y servicios realizados:**

Los sistemas manuales dificultan llevar un registro detallado de servicios previos, preferencias del cliente, frecuencia de visitas o incidencias. Sin esta información, es más complejo ofrecer un trato personalizado o realizar acciones comerciales efectivas.

- **Dificultad para obtener estadísticas precisas de ocupación y demanda:**

Sin una base de datos centralizada y sin herramientas de análisis, resulta prácticamente imposible medir la carga de trabajo, la estacionalidad, los servicios más solicitados o las horas punta. Esto limita la toma de decisiones estratégicas y la planificación del negocio.

Estas limitaciones afectan directamente a la **eficiencia operativa**, provocando desorganización, falta de visibilidad del flujo de trabajo y pérdidas de tiempo. Al mismo tiempo, repercuten en la **experiencia del cliente**, que puede percibir desatención, tiempos de espera prolongados o falta de profesionalidad.

Por todo ello, surge la necesidad de **digitalizar, automatizar y centralizar la gestión de citas**, con el fin de optimizar el funcionamiento interno del lavadero y ofrecer un servicio más rápido, transparente y profesional.

3.2. Estudio de soluciones existentes

Para definir la solución tecnológica más adecuada, se realizó un análisis de diversas herramientas y plataformas de gestión de reservas actualmente disponibles en el mercado. El objetivo fue evaluar si alguna de ellas podía adaptarse directamente a las necesidades de los lavaderos, tomar como referencias y saber qué mejorar en nuestro propio proyecto.

Los criterios analizados fueron:

- **Soluciones de código abierto vs. software propietario:**

Se valoró la libertad de personalización, los costos asociados y la capacidad de modificar o ampliar funcionalidades según las necesidades del negocio.

- **Facilidad de uso y accesibilidad multiplataforma:**

Muchas herramientas ofrecen aplicaciones web o móviles, pero no siempre permiten una integración fluida con sistemas de escritorio o con procesos internos específicos.

- **Flexibilidad para adaptarse a los procedimientos concretos del lavadero:**

Algunas plataformas están pensadas para peluquerías, talleres o servicios generales, y no contemplan particularidades propias de un lavadero, como tiempos variables según el tipo de servicio, vehículos especiales o gestión dinámica de disponibilidad.

- **Capacidad de integración con otros sistemas:**

Se valoró la posibilidad de conectar las plataformas con sistemas de pago, CRM, inventarios o herramientas internas del negocio.

Tabla comparativa de soluciones analizadas

APP	Características principales	Ventajas	Limitaciones
 Picktime	<ul style="list-style-type: none"> • Reservas online 24/7 • Gestión de personal • Opciones de pago integradas 	<ul style="list-style-type: none"> • Accesible a cualquier hora • Configuración sencilla • Adecuado para negocios pequeños y medianos 	<ul style="list-style-type: none"> • Personalización limitada • Algunas funciones avanzadas requieren pago • Adaptación genérica, no específica para lavaderos
	<ul style="list-style-type: none"> • Disponibilidad en tiempo real • Recordatorios automáticos • Pagos integrados 	<ul style="list-style-type: none"> • Plataforma robusta y estable • Automatización de avisos • Integración con múltiples servicios 	<ul style="list-style-type: none"> • Coste mensual elevado según volumen • Curva de aprendizaje moderada • Poca flexibilidad para flujos de trabajo específicos
 EasyWeek	<ul style="list-style-type: none"> • Calendario accesible desde cualquier dispositivo • Gestión de horarios • Sistema multiusuario 	<ul style="list-style-type: none"> • Muy accesible y sencillo de usar • Interfaz moderna • Adecuado para equipos distribuidos 	<ul style="list-style-type: none"> • Personalización limitada • Funciones avanzadas en planes superiores • No pensado para servicios con tiempos variables como lavaderos
 Vev	<ul style="list-style-type: none"> • Pensado para servicios móviles • Planificación inteligente • Recordatorios automáticos 	<ul style="list-style-type: none"> • Ideal para negocios dinámicos • Buena automatización y avisos • Diseño minimalista 	<ul style="list-style-type: none"> • Enfoque no específico para lavaderos • Escalabilidad limitada • Integración reducida con sistemas externos
 urable / Zenbooker / MioCommerce	<ul style="list-style-type: none"> • CRM integrado • Gestión de clientes, citas y pagos • Funciones de marketing 	<ul style="list-style-type: none"> • Más completas a nivel comercial • Seguimiento detallado del cliente • Incluyen herramientas de fidelización 	<ul style="list-style-type: none"> • Coste elevado en la mayoría • Configuración más compleja • Requieren adaptación manual para procesos específicos

4. DISEÑO BACKED

4.1. Diseño General: Arquitectura y Capas

El proyecto consiste en el desarrollo de un Sistema Integral de Gestión de Citas (SGC) para un lavadero de vehículos, implementado bajo una Arquitectura de Servicios Distribuidos con un backend centralizado.

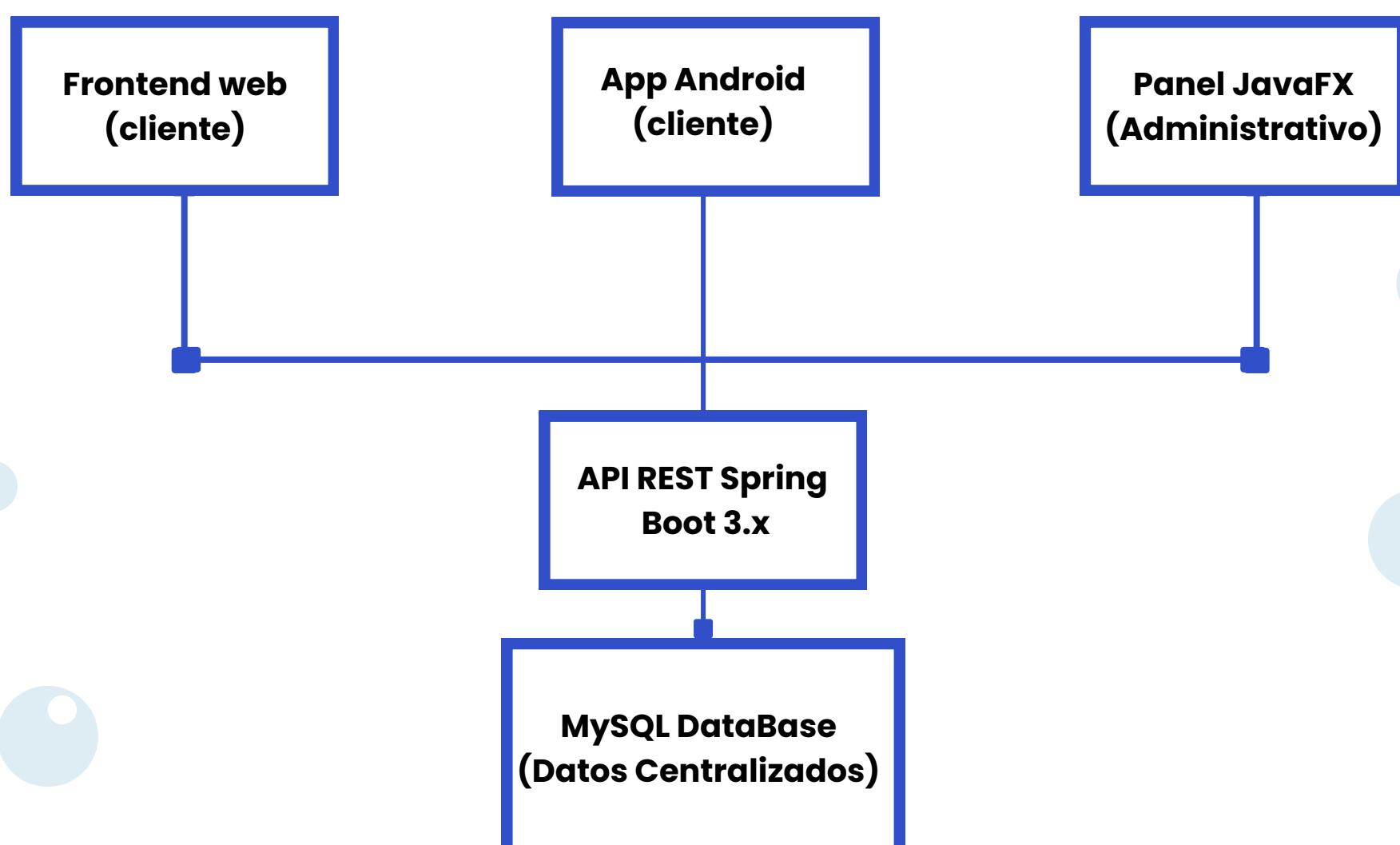
La solución adopta un API-First Design, donde la API RESTful de Spring Boot actúa como la única Capa de Lógica de Negocio (BLL) y el punto de acceso para todas las interfaces de usuario.

Modelo Arquitectónico de Cuatro Capas

El sistema se estructura en un modelo de cuatro capas que garantiza la escalabilidad, la mantenibilidad y la coherencia de los datos en tiempo real:

- **Capa de Presentación Policanal (Frontends):** Múltiples interfaces adaptadas al entorno (Web, Móvil, Escritorio) para cubrir las necesidades de clientes y administradores.
- **Capa de Servicios de Aplicación (API REST):** Responsable de la validación, la lógica transaccional, la Clasificación Inteligente de Vehículos y la integración con servicios externos (SMTP para emails, Google Services).
- **Capa de Persistencia (BBDD):** MySQL 8.x como fuente única de verdad (Single Source of Truth) para el almacenamiento de datos relacionales (citas, tarifas, clientes, logs).
- **Capa de Infraestructura (DevOps):** Uso de Docker y Docker Compose para la containerización, asegurando la portabilidad y el despliegue coherente del stack completo.

Diagrama Conceptual



4.2. Diseño Detallado: Implementación Técnica

4.2.1. Arquitectura de Componentes y Flujo de Datos

Se define el rol específico de cada componente, las tecnologías utilizadas y su propósito dentro del ciclo de vida de la aplicación.

Componente	Rol en el Sistema	Tecnologías Clave	Protocolo de Comunicación	Propósito y Contribución
Frontend Web	Capa de Presentación Pública	Thymeleaf, HTML5, CSS Grid/Flexbox	HTTP(S) hacia API REST	Provee el Portal de Reservas público. Diseño Mobile-First y responsive.
App Android	Capa de Presentación Móvil Nativa	Java Nativo, Retrofit	HTTP(S) hacia API REST	Ofrece una Experiencia de Usuario (UX) optimizada en móvil. Retrofit gestiona las peticiones HTTP.
Panel JavaFX	Capa de Presentación Administrativa	JavaFX, Scene Builder	HTTP(S) hacia API REST	Herramienta de escritorio para la gestión avanzada (CRUD de citas, tarifas y horarios).
API RESTful	Backbone: Lógica de Negocio Centralizada	Spring Boot 3.x (MVC, Data JPA), Java Mail	JDBC hacia BBDD	Centraliza la Inteligencia de Negocio (Clasificación Vehicular, Scheduler) y es el único punto de contacto con la BBDD.
MySQL Database	Capa de Persistencia (Single Source of Truth)	MySQL 8.x	JDBC (JPA)	Almacena de forma relacional y persistente toda la información crítica del sistema.
Docker / Compose	Capa de Infraestructura (DevOps)	Docker, Docker Compose	N/A	Facilita la containerización y el despliegue del stack completo en un único comando, asegurando la consistencia del entorno.

4.2.2. Diseño de Lógica Central (Funcionalidades Clave)

A. Clasificación Automática de Vehículos

Implementación: Módulo dentro de la API REST (Service Layer).

- Objetivo: Determinar automáticamente la tarifa y el tiempo de servicio requerido al recibir el tipo de vehículo del cliente.
- Mecanismo: El frontend envía la información y la API aplica la lógica condicional (ej: Turismo) para devolver la categoría de servicio final.

B. Sistema de Notificaciones Asíncronas

- Tecnología: Java Mail orquestado por la API REST.
- Funcionalidades: Envío de confirmaciones transaccionales inmediatas y activación de un Scheduler de Spring para el envío programado de recordatorios de cita (ej. 24h antes), minimizando ausencias (no-shows).

C. Gestión Flexible de Horarios y Disponibilidad

- Lógica: La API valida las solicitudes de reserva consultando en tiempo real la MySQL Database.
- Propósito: Garantizar que el slot solicitado por el cliente esté libre, considerando la duración variable del servicio según la clasificación del vehículo.