



PROYECTO 1

INTEGRACIÓN DE APLICACIONES

MARINA AMAYA, IGNACIO FERNÁNDEZ Y
BRITNEY CUICHAN

Índice

Introducción	1
Contexto del proyecto	1
1. Actividad principal	1
2. Sistemas de Información existentes	1
Sistemas de negocio	1
Sistemas operacionales y IoT fragmentado.....	2
3. Problema de integración a resolver	2
El problema en acción	2
Objetivo del proyecto de integración	3
Modelos de datos común.....	3
Identificación y función de entidades	3
Entidades administrativas.....	3
Entidades operacionales (IoT / Mantenimiento)	3
2. Relación entre entidades	4
3. Decisiones Clave	4
Enfoque de integración (IoT y Arquitectura)	4
Justificación de campos específicos.....	5
Arquitectura inicial	5
Diagrama de componentes	5
Descripción general.....	5
Componentes principales.....	5
Flujos de comunicación.....	6
Ventajas de esta arquitectura	6
Diagrama de Contenedores	7
Diagrama Lógico	8
Relaciones entre entidades.....	8
Propósito del diseño lógico	8
Conclusión	9

Introducción

El proyecto del Tema 1 de Integración de Aplicaciones tiene como objetivo establecer la base de un sistema donde diferentes servicios puedan trabajar juntos y compartir información de forma unificada.

A través de un caso práctico, se busca mostrar cómo conectar sistemas como un CRM, un ERP y una red de sensores IoT mediante una arquitectura que deja gestionar y analizar los datos de manera centralizada.

Para desarrollar esta parte, se eligió el proyecto de WinterTrack que es una empresa dedicada a la gestión de una estación de esquí, que sirve como ejemplo para aplicar los conceptos de integración y diseño de arquitectura.

Contexto del proyecto

WinterTrack es una estación de esquí que busca la digitalización integral de su gestión, con un enfoque en la mejora de la experiencia del cliente y la optimización de los recursos operativos mediante sensores IoT.

1. Actividad principal

WinterTrack opera una estación de esquí centrada en los deportes de invierno. Sus actividades clave incluyen:

- **Operación y seguridad:** Mantenimiento y gestión segura de las pistas de esquí y de toda la infraestructura de transporte (telesillas, telecabinas, cintas).
- **Servicios al cliente:** Venta de forfaits, alquiler de material, escuela de esquí y gestión de servicios de restauración.
- **Gestión de recursos:** Control de consumos energéticos, optimización de inventarios y uso eficiente del agua para la producción de nieve artificial.

2. Sistemas de Información existentes

Actualmente, los sistemas de WinterTrack operan como segmentación de datos, lo que impide una visión entera del negocio:

Sistemas de negocio

- **CRM:**
 - Gestiona la base de datos de clientes, el historial de forfaits y las reservas.

- **Fallo:** No tiene acceso a datos de uso de remotes o estados de operación en tiempo real.
- **ERP:**
 - Sistema para la gestión financiera, compras, inventario y RR.HH.
 - Fallo: No está conectado a los datos operativos de mantenimiento o a los consumos energéticos de la infraestructura.

Sistemas operacionales y IoT fragmentado

- **Sensores de pistas:** Estaciones meteorológicas y sensores de nieve.
 - Fallo: Solo ofrecen una visualización básica y los datos se exportan manualmente una vez al día.
- **Controladores de remotes (PLCs):** Gestionan velocidad y estado de los remotes.
 - Fallo: Los registros de fallos solo son accesibles desde la sala de control local.
- **Sistema de Ticketing (Acceso):** Tornos que leen los forfaits.
 - Fallo: La información de paso de cliente (aforo) se procesa al final del día, sin impacto inmediato en la gestión de colas.

3. Problema de integración a resolver

El principal desafío es la falta de una visión unificada y en tiempo real entre los datos operacionales (IoT) y los datos de negocio (CRM/ERP).

El problema en acción

Cuando un sensor detecta una alta vibración en un telesilla:

- **No hay comunicación al ERP:** No se genera automáticamente una orden de trabajo de mantenimiento ni se verifica la disponibilidad de repuestos.
- **No hay comunicación al CRM:** No se alerta a los clientes en la aplicación móvil sobre un posible retraso o cierre.
- **No hay cruce de datos:** No se puede usar la información de aforo para determinar el impacto real en los esquiadores y redirigirlos activamente a otras zonas.

Objetivo del proyecto de integración

El proyecto busca integrar los datos de los sensores IoT (nieve, temperatura, vibración, aforo) en una Plataforma Central de Datos (API Unificada). Esta plataforma permitirá que todos los sistemas consuman la información en tiempo real para:

- **Mantenimiento predictivo:** Activar órdenes de trabajo automáticas en el ERP.
- **Mejora del cliente:** Informar en tiempo real a los esquiadores sobre estados de remotes y aforos.
- **Optimización de recursos:** Usar datos de nieve y temperatura en tiempo real para optimizar los sistemas de innovación.

Modelos de datos común

Se describe las entidades clave y sus relaciones, estableciendo el estándar de datos que utilizará la API Unificada para integrar la información de los sistemas CRM, ERP y el IoT fragmentado.

Identificación y función de entidades

Se clasificaron las entidades en dos grupos principales para reflejar los sistemas de origen:

Entidades administrativas

Estas entidades gestionan los aspectos comerciales y de servicio al cliente.

- **Cliente:**
 - Tipo de sistema: CRM.
 - Función: Gestiona la información del usuario, el tipo de forfait y el saldo precargado. Es clave para la personalización de la experiencia y la seguridad.
- **Pedido:**
 - Tipo de sistema: ERP.
 - Función: Registra las transacciones financieras y comerciales (compra de forfait, alquiler). Es esencial para el control de inventario y facturación.

Entidades operacionales (IoT / Mantenimiento)

Estas entidades capturan la información del entorno físico y la infraestructura.

- **Remonte:**
 - Tipo de sistema: Operacional/IoT (Infraestructura).
 - Función: Representa la instalación física del remonte. Su propósito es contextualizar las lecturas del sensor y permitir el monitoreo de su estado general.
- **LecturaSensor:**
 - Tipo de sistema: Operacional/IoT (Datos).
 - Función: Entidad fundamental para la monitorización en tiempo real. Captura el dato crudo (vibración, temperatura, aforo) junto con su marca de tiempo.

2. Relación entre entidades

El modelo se basa en referencias de identificadores (string de ID) que permiten a la API Unificada consolidar la información.

Relación 1: Cliente -> Pedido (1:N)

- **Explicación:** Un **Cliente** (Cliente.clienteld) puede generar Múltiples Pedidos a lo largo del tiempo.
- **Mecanismo:** El campo Pedido.clienteld actúa como la clave foránea, referenciando a Cliente.clienteld.

Relación 2: Remonte -> LecturaSensor (1:N)

- **Explicación:** Un **Remonte** (Remonte.remonteld) genera Múltiples LecturasSensor (datos) en alta frecuencia.
- **Mecanismo:** El campo LecturaSensor.remonteld actúa como clave foránea, referenciando a Remonte.remonteld.

3. Decisiones Clave

Enfoque de integración (IoT y Arquitectura)

- **Separación de Infraestructura y Datos:** Se eligió separar la entidad Remonte (estable) de LecturaSensor (alto volumen y alta frecuencia). Esto permite que el tráfico de datos crudos no sature la información estática del remonte y facilita el mantenimiento predictivo.

- **Campos de relación:** Se usan IDs de referencia como `remonteld` y `clienteld` para facilitar la consulta cruzada en la API Unificada, que es el objetivo central del proyecto.

Justificación de campos específicos

- **Cliente.saldoMonedero:** Incluimos este campo para permitir la integración futura de servicios de pago rápido (monedero virtual), fusionando datos de fidelización (CRM) con puntos de venta (ERP).
- **Pedido.estado (uso de enum):** Se define un conjunto limitado de estados (ej. "Completado", "Pendiente de Pago") para estandarizar la comunicación del ciclo de vida de la transacción a través de todos los sistemas.
- **Remonte.estadoOperacional (uso de enum):** Este campo es crítico para la integración. Es el dato que debe ser actualizado en tiempo real (basado en `LecturaSensor` y mantenimientos) y consumido por la aplicación del cliente (CRM/App) para informar sobre cierres o retrasos.
- **LecturaSensor.fechaHora (Formato ISO 8601):** El uso del formato "date-time" estandarizado es vital para asegurar que todas las fuentes de datos (sistemas antiguos, nuevos sensores) utilicen el mismo huso horario y formato, previniendo errores de sincronización en el análisis de eventos.

Arquitectura inicial

Diagrama de componentes

Descripción general

La arquitectura está diseñada para ser simple, funcional y escalable. La API unificada actúa como punto central de conexión entre usuarios, CRM e IoT, organizando y combinando los datos para ofrecerlos listos para usar.

Componentes principales

- **API Unificada:** núcleo del sistema. Recibe peticiones de usuarios (cliente, empleado, gestor) y de la interfaz web. Coordina los servicios y entrega datos consolidados.
- **CRM:** gestiona información administrativa como clientes, pedidos, facturas y pagos.
- **IoT:** proporciona datos operativos desde sensores y máquinas (remontes, lecturas, temperaturas, vibraciones).

- **Base de Datos:** almacena la información combinada para consultas y reportes.
- **Interfaz Web / Postman:** punto de acceso para visualizar la información unificada.

Flujos de comunicación

- Comunicación vía REST sobre HTTP, con datos en formato JSON.
- La API realiza llamadas como:
 - GET /clientes, GET /pedidos al CRM
 - GET /lecturas, GET /remotes al IoT
- Usuarios interactúan con la API mediante la web o Postman.

Ventajas de esta arquitectura

- **Centralización:** todos los datos pasan por la API, que los combina y entrega.
- **Mantenimiento sencillo:** cambios en CRM o IoT no afectan a clientes ni interfaz.
- **Control y seguridad:** validación, autenticación y autorización centralizadas en la API.

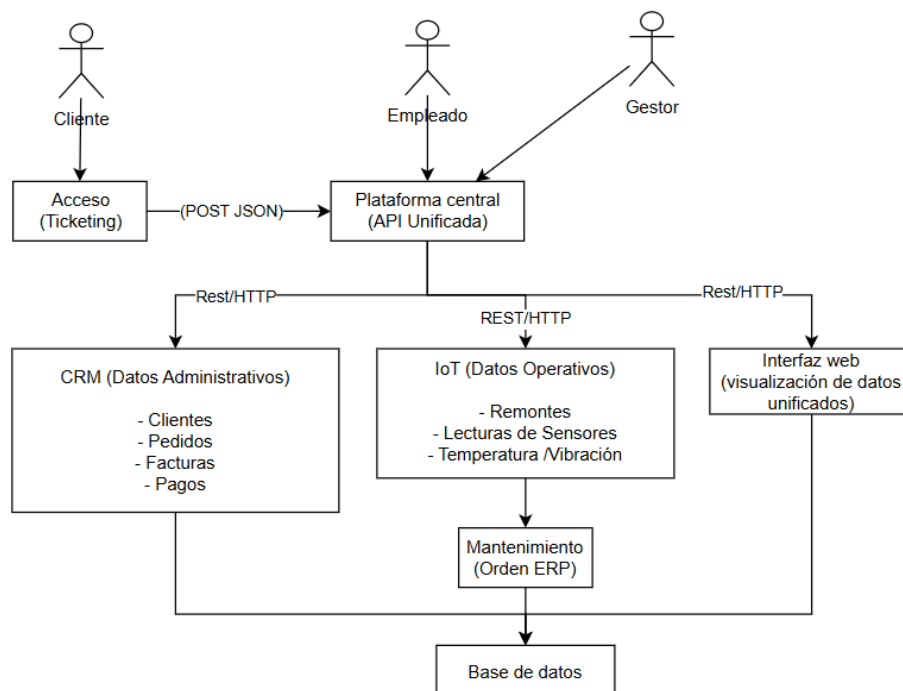


Diagrama de Contenedores

Representa los principales bloques de software del sistema.

API Unificada

Conecta y coordina los servicios, procesando peticiones de usuarios.

Servicios conectados

- **CRM:** datos administrativos.
- **IoT:** datos de sensores y remontes.
- **Ticketing:** control de accesos y autenticación.

Base de datos

Centraliza la información combinada de todos los servicios.

Sistemas externos

- **Pasarela de pagos** (Stripe, PayPal): procesamiento de transacciones.
- **ERP:** gestión de órdenes de mantenimiento generadas por IoT.

Acceso de usuarios

- **Navegador Web:** para gestores.
- **Aplicación Cliente:** para clientes y empleados.

Comunicación

- REST sobre HTTP
- Formato JSON

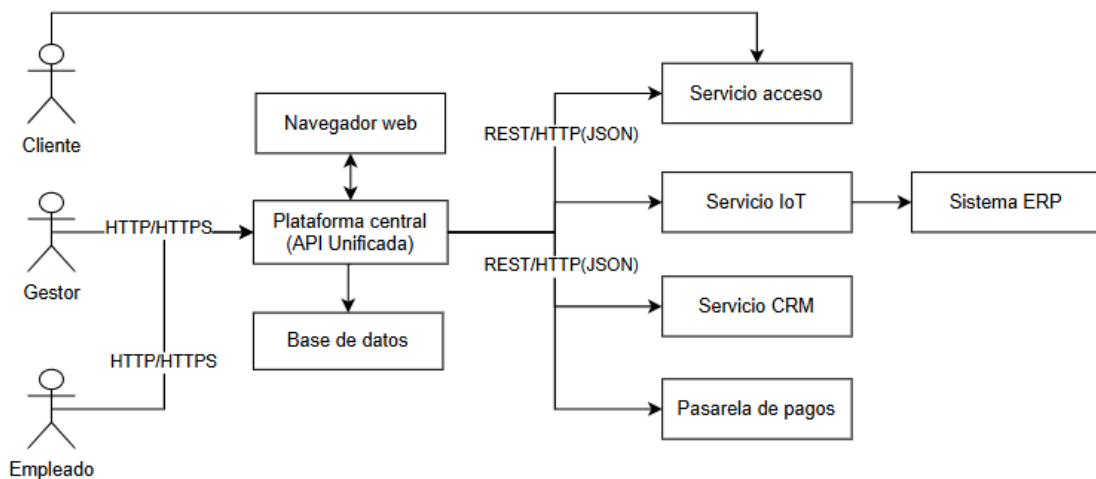


Diagrama Lógico

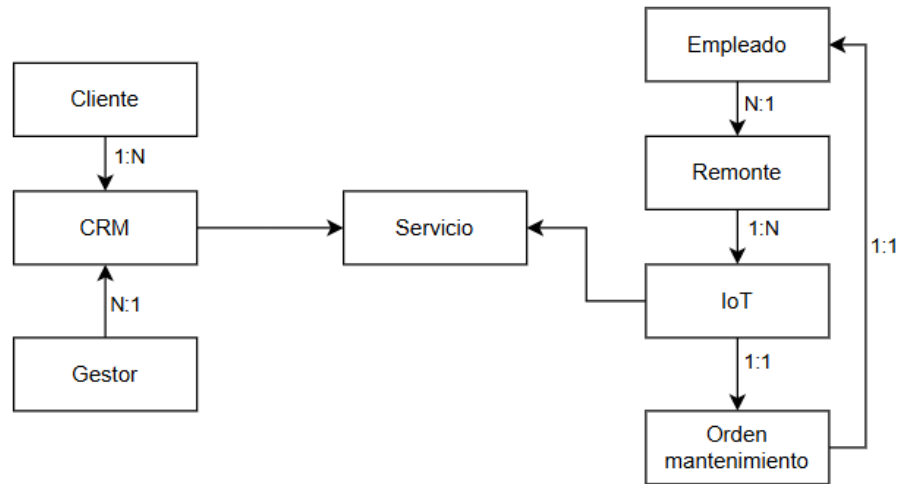
Representa las relaciones entre entidades del sistema.

Relaciones entre entidades

- Un cliente puede estar vinculado a múltiples registros en el CRM, lo que permite gestionar varias interacciones, pedidos o facturas por cada cliente (relación 1:N).
- Cada registro en el CRM está gestionado por un único gestor, aunque un mismo gestor puede encargarse de múltiples clientes o cuentas (relación N:1).
- El CRM está directamente conectado con el servicio, permitiendo que la información administrativa fluya hacia los procesos operativos sin intermediarios.
- El servicio se vincula directamente con el sistema IoT, lo que permite asociar datos técnicos y lecturas de sensores con los servicios prestados.
- Cada empleado puede estar asignado a un único remote, que representa un punto de control o dispositivo físico, aunque varios empleados pueden compartir el mismo (relación N:1).
- Un remonte puede estar conectado a varios dispositivos IoT, permitiendo la supervisión y control de múltiples sensores desde un único punto (relación 1:N).
- Cada dispositivo IoT genera una única orden de mantenimiento, lo que asegura trazabilidad directa entre el estado del sensor y la acción correctiva (relación 1:1).
- Finalmente, cada orden de mantenimiento está asignada a un único empleado, cerrando el ciclo operativo con responsabilidad clara sobre la ejecución de la tarea (relación 1:1).

Propósito del diseño lógico

Visualizar cómo se estructuran y relacionan usuarios, servicios y dispositivos para facilitar la gestión y el análisis de datos.



Conclusión

Con este primer tema se estableció las bases del proyecto de integración, definiendo el contexto, los sistemas implicados y la arquitectura general. A través del caso de WinterTrack se pudo entender cómo diferentes servicios como CRM, ERP e IoT pueden conectarse para compartir datos y mejorar la gestión de una empresa. Este proyecto facilitó la aplicación práctica de los conceptos teóricos, mediante la creación de un modelo de datos unificado y la representación visual de la estructura del sistema.