

Prueba Técnica para Desarrollador Backend

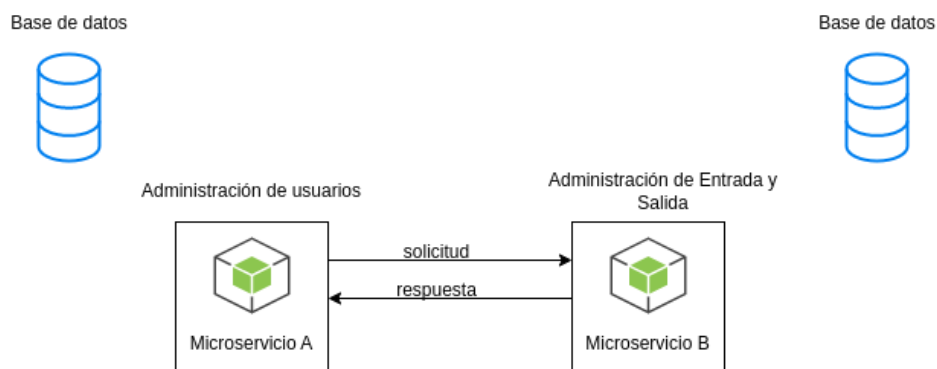
Ejercicio 1.

La empresa **Pentview** necesita un sistema de control de asistencia que permita llevar el registro de entradas y salidas de sus empleados. Este sistema debe seguir una arquitectura basada en microservicios, usando comunicación asincrónica para validar datos entre servicios:

- Diseñar e implementar una solución back-end distribuida en microservicios.
- Registrar entradas y salidas de empleados con validación mediante mensajería asincrónica (RabbitMQ).
- Implementar prácticas de desarrollo modernas: separación de responsabilidades, validaciones, control de errores, documentación, etc.

Datos Técnicos

El sistema debe estar compuesto por **dos microservicios independientes**, con bases de datos separadas:



Microservicio A.

- CRUD completo de empleados (crear, listar, actualizar, eliminar).
- Validar existencia de empleados por identificador recibido desde otro servicio (vía RabbitMQ).
- Base de datos independiente.
- Documentación Swagger (opcional pero recomendable).

Microservicio B.

- Registro de entradas y salidas de empleados.
- Un registro contiene: fecha, hora, tipo (entrada/salida), idEmpleado.
- Validar al empleado antes de registrar usando RabbitMQ (enviar mensaje a Microservicio A y esperar respuesta).
- Base de datos independiente.
- Control de duplicación (por ejemplo, evitar dos entradas seguidas sin salida).

Los microservicios A y B, tendrán una base de datos independiente y sus respectivos endpoints.

Ejercicio 2.

Basado en el **Ejercicio 1**, se debe implementar una comunicación asíncrona utilizando el Message Broker **RabbitMQ**, esta comunicación será utilizada por el **Microservicio B**, desde donde se enviará el identificador del empleado al cual se le va a registrar una entrada o salida, dicho identificador será recibido por el **Microservicio A**, el cual validará que el identificador recibido corresponda a un empleado registrado en su base de datos, el **Microservicio A** deberá responder afirmativa o negativamente, dependiendo de esta respuesta, el **Microservicio B** registrará o no la entrada o salida.

Tecnologías.

- Lenguajes: Node.js (NestJS) o Java (Spring Boot).
- Bases de datos: PostgreSQL, MySQL, o MongoDB.
- Mensajería: RabbitMQ (obligatorio).
- Versionado: Git (de preferencia en GitHub, GitLab o Bitbucket).

Entregables.

- Código fuente de los microservicios en un repositorio o archivo ZIP.
- Instrucciones para ejecutar los microservicios (README).
- Archivo docker-compose.yml funcional (si aplica).
- Postman Collection o documentación Swagger para probar endpoints.
- (Opcional) Diagrama de arquitectura o flujo de comunicación.

Indicaciones.

- La prueba tendrá una duración de 3 días, a partir del envío de la misma al correo del aplicante.
- Si el aplicante lo ve necesario puede ampliar los requisitos (puntos extras).
- Se puede solicitar información extra, si al aplicante no le ha quedado claro los requisitos del reto.
- Cualquier duda puede ser consultada al correo manuel.a@pentview.com o al teléfono [0985576629](tel:0985576629).



Pentview