

Enunciado

Construye un **servicio “Incidentes y Solicitudes”** (API REST) para registrar y dar seguimiento a incidencias de aplicaciones internas.

Requerimientos funcionales (mínimos)

1. **CRUD de tickets** con campos: id, titulo, descripcion, tipo (incidente/solicitud), aplicacion, prioridad, estado (abierto, en_progreso, resuelto, cerrado), creado_en, actualizado_en, creado_por, asignado_a.
2. **Listado con filtros** por estado, aplicacion, rango de fechas y prioridad; **paginación** y **ordenamiento**.
3. **Transición de estados** con reglas simples (p.ej., no se puede cerrar si no está resuelto).
4. **Comentarios** por ticket (hilo cronológico).
5. **Validación** de entrada y **manejo de errores** consistente (códigos HTTP, mensajes claros).

Requerimientos técnicos (mínimos)

- **Modelo relacional** (migraciones SQL), índices básicos y claves foráneas.
- **Pruebas** unitarias de servicios/lógica y al menos 1 prueba de integración (endpoint feliz y uno con error).
- **README** con instrucciones para correr, supuestos y trade-offs; **diagrama** (p. ej., ER o de componentes).
- **Registro** (logging) básico y trazabilidad de una petición.

Extensiones

- **Consultas/analítica:** endpoint de reportes con **MTTR** (tiempo promedio de resolución) por aplicación y **top 3 aplicaciones** con más incidentes en 30 días.
- **Optimización:** justificar índices, demostrar un **plan de ejecución** antes/después u optimización de N+1.
- **Diseño:** breve propuesta para **escalar** (caching, colas para notificaciones, particionamiento por aplicación, idempotencia para reintentos).
- **Observabilidad:** métricas (contadores de 5xx/4xx por endpoint) y alerta textual propuesta.
- **Seguridad:** RBAC simple (roles: user, agent, admin) aplicado a endpoints críticos.

Detalle de entregables

- **Código** en un repo privado (estructura por capas).
- **Migraciones/Script SQL** (con índices) + **seeds** mínimos.
- **Colección de pruebas** (unit/integración) que corran con un solo comando.
- **README** (setup, decisiones, supuestos, límites) + **diagrama** (ER o componentes).
- **Postman/HTTP file** con ejemplos de requests.
- **Breve documento** (½–1 página) con métricas/monitoreo y, si aplica, RBAC.

Restricciones Importantes

- Debe entregarse de manera totalmente funcional, con todo lo necesario para poder ejecutarse y probarse adecuadamente.
- No utilizar herramientas IA para generarlo, disponemos de una herramienta para detectar el uso de IA de desarrollo.
- Es posible que no puedas entregar todo lo que se te pide, lo importante es que hagas funcione perfectamente y sin errores, que demuestres tus habilidades y capacidades. Es posible que tengas que hacernos una defensa de tu propuesta en una reunión.