

# Analisi Backend Node/Express in TypeScript – Enterprise Ready

## 1. Premessa

Sei incaricato della gestione di un backend performante in Node.js con Express e TypeScript. Il sistema deve

## 2. Architettura attuale (ipotetica)

Presunto monolite modulare o microservizio. Express come framework, DB relazionale o NoSQL, container D

## 3. Requisiti principali

- Modularità per domini/applicativi
- Performance elevate
- Logging, tracing e metrics
- Manutenibilità (test, tipi, lint)
- Sicurezza (JWT, CORS, header)
- Testing automatico + CI/CD

## 4. Strategia tecnica

- Modularizzazione codice (cartelle per modulo)
- DI container (tsyringe, inversify)
- Logging avanzato con Pino
- Tracing con OpenTelemetry
- Sicurezza con Helmet, JWT, validazione input
- API versioning e documentazione Swagger

## 5. Considerazioni UX/API

- Risposte coerenti (data, error, meta)
- Versioning API (/v1, /v2)
- Circuit breaker per fallback
- API idempotenti, sicure e tracciabili

## 6. Alternative & trade-off

Express: stabile ma meno veloce

Fastify: schema-based, più performante

Monolito: semplice da mantenere

Microservizi: scalabili ma complessi

Zod vs Joi per validazione

Prisma ORM per type safety

## 7. Roadmap

Fase 1 – Audit e analisi applicativi

Fase 2 – Definizione standard modulare

Fase 3 – Logging, tracing, sicurezza

Fase 4 – CI/CD, container, monitoring