

# Diseño de Alto Nivel en el Desarrollo de Software



## ¿Qué es el Diseño de Alto Nivel?

El **Diseño de Alto Nivel (HLD - High Level Design)** es una etapa del desarrollo de software donde se define **cómo estará estructurado el sistema**, sin entrar aún en detalles de programación.

Es como hacer un plano general de una casa: defines cuántos cuartos tendrá, dónde están, cómo se conectan entre sí, y qué materiales usarás, **pero no decides todavía cómo será cada puerta o enchufe**.



## Objetivo del Diseño de Alto Nivel

- Traducir los **requerimientos funcionales y no funcionales** en una **arquitectura técnica** comprensible.
- Proporcionar una visión clara de **cómo se dividirá el sistema, qué tecnologías se usarán, y cómo interactuarán los componentes**.



## Elementos Clave del Diseño de Alto Nivel

A continuación, los elementos que debes definir en esta fase:

### 1. Arquitectura del sistema

- ¿Qué tipo de arquitectura tendrá el software?
  - **Monolítica**
  - **Cliente-servidor**
  - **Microservicios**
  - **Arquitectura en capas**
  - **Serverless**, etc.



*Ejemplo:* “El sistema será una aplicación web cliente-servidor basada en microservicios.”

### 2. Componentes principales del sistema

- ¿Cuáles son las partes grandes del sistema?
  - Módulo de autenticación
  - Módulo de pagos
  - Módulo de gestión de usuarios
  - API de productos, etc.

✓ *Ejemplo:* “El sistema tendrá un módulo de usuarios, un backend REST y una base de datos relacional.”

### 3. Interacción entre componentes

- ¿Cómo se comunicarán entre sí?
  - HTTP, API REST, gRPC, colas de mensajes (RabbitMQ, Kafka)
  - Eventos, sockets, etc.

✓ *Ejemplo:* “El frontend se comunicará con el backend mediante API REST, y los microservicios usarán colas de mensajes para eventos internos.”

### 4. Selección de tecnologías

- Lenguajes de programación (Python, Java, etc.)
- Frameworks (React, Spring Boot, etc.)
- Base de datos (MySQL, MongoDB, etc.)
- Servicios en la nube (AWS, Azure, etc.)

✓ *Ejemplo:* “Se usará React para el frontend, Node.js para el backend, y PostgreSQL como base de datos.”

### 5. Diagramas de arquitectura

Se suelen incluir uno o más diagramas visuales para explicar la arquitectura general del sistema.

👉 Puedes usar:

- **Diagrama de componentes**
- **Diagrama de despliegue**
- **Modelo C4 (Contexto, Contenedor, Componente, Código)**

 **Herramientas recomendadas:** Lucidchart, Draw.io, Visual Paradigm, PlantUML.



## Resultado del Diseño de Alto Nivel

Al final de esta fase deberías tener:

- Un documento o presentación con la arquitectura del sistema.
- Diagramas explicativos.
- Lista de tecnologías y herramientas a usar.
- Definición de los principales módulos del software y sus interacciones.



## Analogía práctica

El **diseño de alto nivel** es como hacer el mapa de un videojuego: defines el mundo, los niveles, los enemigos y las armas principales... pero no dibujas aún cada textura o animación.

Eso vendrá en el **diseño de bajo nivel**.



## Recomendaciones

- No te vayas al detalle técnico; céntrate en la **visión global**.
- Valida el diseño con el equipo (arquitectos, desarrolladores, stakeholders).
- Asegúrate de que el diseño cumple con los **requisitos no funcionales** (seguridad, rendimiento, escalabilidad, etc.)