

## **UNIDAD TEMÁTICA 6 – Patrones y arquitectura – Trabajo final de unidad.**

Cada uno de estos temas se dividirá en los equipos y cada uno hará una presentación al resto de la clase.

### **Arquitectura en capas y cliente-servidor:**

- Key Concepts: Definir arquitectura en capas (por ejemplo, modelo de 3 capas) y arquitectura cliente-servidor.
- En qué caso aplico una frente a la otra: Analizar escenarios donde sea más adecuado usar arquitectura en capas o cliente-servidor.
- Ventajas y desventajas: Presentar pros y contras de cada arquitectura.
- ¿Se pueden combinar?: Explicar cómo estas arquitecturas pueden ser usadas juntas y en qué escenarios sería beneficioso.

### **Arquitectura monolítica vs microservicios:**

- Key Concepts: Definir arquitectura monolítica y arquitectura de microservicios.
- En qué caso aplico una frente a la otra: Analizar casos de uso donde una arquitectura monolítica sea más adecuada en comparación con microservicios y viceversa.
- Ventajas y desventajas: Listar los pros y contras de ambas arquitecturas.
- ¿Se pueden combinar?: Explorar si es posible y conveniente utilizar una estrategia híbrida combinando elementos de ambas arquitecturas.

### **Arquitectura orientada a eventos y SOA (Arquitectura Orientada a Servicios):**

- Key Concepts: Definir Arquitectura Orientada a Eventos (EDA) y Arquitectura Orientada a Servicios (SOA).
- Ventajas y desventajas: Comparar los beneficios y limitaciones de EDA y SOA.
- Realizar demo utilizando RabbitMQ sobre EDA: Crear una demostración práctica que muestre cómo implementar una Arquitectura Orientada a Eventos utilizando RabbitMQ.
- ¿Se pueden combinar?: Explorar la integración de SOA y EDA y cómo esto puede ser beneficioso.

### **Cache - Implementar parte del proyecto con Redis:**

- Key Concepts: Definir qué es un cache y cómo Redis puede ser utilizado para el almacenamiento en caché.
- Realizar demo sobre alguna parte del proyecto donde aplique (u otra aplicación).
- Ventajas y desventajas: Discutir los beneficios y las limitaciones de utilizar Redis para el almacenamiento en caché.

### **GraphQL/gRPC - conceptos y demo:**

- Key Concepts: Definir GraphQL y gRPC, y explicar cómo se utilizan en el desarrollo de aplicaciones.
- Realizar demo sobre alguna parte del proyecto donde aplique (u otra aplicación).
- Ventajas y desventajas: Presentar los pros y contras de utilizar GraphQL y gRPC.

Cada equipo deberá entregar una presentación donde incluya al menos estos ítems:

- Introducción
- Desarrollo del problema
- Conclusiones
- Bibliografía