1. **EXPLIQUE COM SUAS PALAVRAS O QUE É DOMAIN DRIVEN DESIGN E SUA IMPORTÂNCIA NA ESTRATÉGIA DE DESENVOLVIMENTO DE SOFTWARE.**

É um conjunto de princípios para desenvolvimento de software baseado em domínio. No mundo onde as transformações acontecem rapidamente e as mudanças de escopo são recorrente, o DDD cai como uma luva devido que sua camada de domínio ser totalmente independente das outras camadas, podendo ser rapidamente adaptada. Alinha o código com o negócio aumentando o entendimento de todos os envolvidos.

1. **EXPLIQUE COM SUAS PALAVRAS O QUE É E COMO FUNCIONA UMA ARQUITETURA BASEADA EM MICROSERVICES. EXPLIQUE GANHOS COM ESTE MODELO E DESAFIOS EM SUA IMPLEMENTAÇÃO.**

Uma arquitetura baseada em micro serviço é a criação serviços mais enxutos com responsabilidades específicas. Ganhos são a flexibilidade , escalabilidade e manutenção simples. Desafios em sua implementação variam desde do planejamento até custos. É necessário um estudo mais detalhado antes de construir, diferente da arquitetura monolítica. Importante automatizar os deploys e fazer o versionamento devido a evolução de cada micro serviço. Deve-se possuir ferramentas que facilitam o gerenciamentos dos micro serviço e a criação de monitoramentos e logs são essencial para identificar problemas. Por último e não menos importante é a segurança com a comunicação e acesso que devem ser construída por um serviço responsável para isso.

1. **EXPLIQUE QUAL A DIFERENÇA ENTRE COMUNICAÇÃO SINCRONA E ASSINCRONA E QUAL O MELHOR CENÁRIO PARA UTILIZAR UMA OU OUTRA.**

**COMUNICAÇÃO SINCRONA:** Significa que a comunicação entre o emissor e o receptor devem estar sincronizada e na ordem. O cenário ideal para os casos que uma ação deve esperar o retorno de outra ação, utilizando as vezes o retorno da primeira para utilizar na segunda ação ou simplesmente a necessidade de executar na ordem.

**COMUNICAÇÃO ASSINCRONA:** Significa que a comunicação entre o emissor e o receptor não precisam estar sincronizada e podem ser executadas em ordens diferentes. O cenário ideal é quando uma ação não depende da outra para ser executado melhorando o tempo de reposta.