1. **A evolução dos sistemas computacionais sempre ocorreu com a introdução de melhorias nos modelos anteriores. Descreva de forma sucinta os fatores motivadores para a evolução dos seguintes sistemas:**
2. **Década de 60 (Mainframes) -> Década de 70(time sharing)**

Na época dos mainframes os usuários que queriam realizar suas computações utilizavam cartões perfurados e tinham que se encaminhar para o setor de processamento de dados para solicitar a execução de seus cartões/computações. As tarefas dos diversos funcionários eram então executadas em sequencia. Infelizmente, isso não apresentava uma boa interface para os usuários e com a popularização da utilização dos mainframes para fazer as computações tornou o tempo de espera cada vez mais elevado. Introduziu-se então os terminais burros onde os usuários poderiam entrar com os dados das suas computações e o esquema de time sharing, onde as computações eram ainda executadas em sequencia, mas havia uma rotatividade entre as aplicações na fila dando a ideia geral de paralelismo entre as computações.

1. **Década de 80 (Microprocessadores) -> Década de 90(Redes)**

A utilização de discos móveis(disquetes) tornou-se cada vez mais “enfadonha”. Era difícil compartilhar dados entre pessoas quando eles estavam confinados aos disquetes. O advento das redes cuidou exatamente desse fato, agora com as redes o compartilhamento dos dados era mais facilitado. O fato dos arquivos estarem disponíveis na rede permitiu que os arquivos permanecessem atualizados assim q fossem modificados promovendo uma maior consistência.

1. **O modelo Cliente/Servidor é um estilo arquitetural desenvolvido para dar suporte a aplicações em uma rede de computadores. No seu processo evolutivo, o C/S sofreu diversas influencias ao longo do tempo. Descreva as principais características citando pelo menos uma vantagem e uma desvantagem das seguintes ondas no modelo C/S.**
2. **Servidor de Arquivos**

O lado do cliente solicita arquivos e o servidor de arquivos os fornece. Possui como única vantagem o pioneirismo histórico e algumas desvantagens. Uma delas é que a manipulação do arquivo é totalmente feita pelo cliente, além disso a transferência de arquivos muito grande é proibitiva.

1. **SGBD**

Um banco de dados onde pode-se fazer consultas, ou seja, vários usuários podem utilizar o banco ao mesmo tempo contanto que suas consultas não conflitem. Possui como vantagem exatamente o fato de vários usuários poderem utilizá-lo e desvantagem o desempenho num ambiente em rede em aplicações muito complexas seria insuficiente.

1. **O modelo C/S sofreu também influência dos paradigmas de programação, explique como as características encapsulamento e herança influenciarem esse modelo sob o ponto de vista dos sistemas distribuídos.**

Durante toda a evolução dos paradigmas de programação vemos dados e processos serem tratados como entidades que devem vir separadas. O modelo C/S em si quando visto sob uma luz estruturada mostra isso, mas com o advento do paradigma orientado a objeto vemos a entidade base do paradigma, a classe, carregar dados (atributos) e processos(método) em um mesmo local, a instancia/objeto. Entretanto, objetos são entidades de memória e precisam ser persistidos as vezes o que novamente levanta a questão de onde colocar os dados.

O conceito de herança no contexto de sistemas distribuídos não está tão bem associado quanto o encapsulamento visto que “herança não combina com distribuição”. Isso se deve ao fato de dados compartilhados entre classes que herdam de outras classes (variáveis de classe) são de difícil atualização. Uma solução para esse problema é trabalhar (distribuir) ramos inteiros de herança (ou componentes) o que garante a integridade das variáveis de classe.

1. **Como o C/S pode dar suporte à implementação dos Sistemas distribuídos? Explique considerando os aspectos de transparência de acesso e localização.**

Levando em consideração que para implementar o sistema distribuído precisamos ter um sistema fracamente acoplado, no caso do cliente/servidor utiliza-se a rede e que o objetivo do sistema distribuído é mostrar vários computadores independentes como um é preciso implementar várias transparências para que isso corra. As duas excenciais para a existência de um sistema distribuído são a de acesos e de localização. A transparência de acesso fala que o usuário deve acessar arquivos/recursos remotos ou locais como se todos estivessem locais, já a transparência de localização afirma que o usuário não precisa saber a localização física dos recursos. Na luz do cliente servidor o mesmo oferece transparência de localização naturalmente através do endereçamento por servidor de nomes, numa abordagem remota e local e o broadcast numa abordagem local. Para fornecer a transparência de acesso é preciso lançar mão do RPC, uma técnica onde as funções de acesso remoto são empacotadas em stubs e tanto para o usuário como para o servidor as chamadas de tais funções funcionam como se fossem locais.

1. **O que é RPC e como esse influenciou o desenvolvimento dos sistemas distribuídos?**

Remote Procedure Call é uma técnica pela qual um sistema cliente/servidor pode implementar a transparência de acesso. Para implementar um sistema distribuído a arquitetura cliente/servidor precisa implementar as transparências de acesso, através do RPC por exemplo e de localização, algo que ela já faz através do servidor de nomes.