***Arquitetura de Sistemas***

**Monolítica**

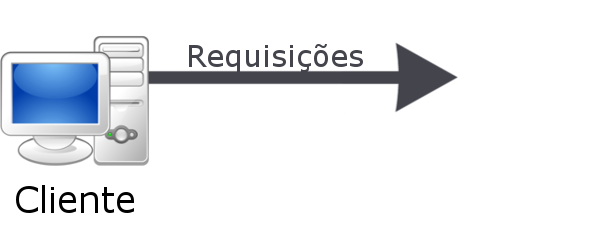
Arquitetura Monolítica é um sistema único, não dividido, que roda em um único processo, uma aplicação de software em que diferentes componentes estão ligados a um único programa dentro de uma única plataforma, assim, os módulos compartilham recursos de processamento, memória, bancos de dados e arquivos.

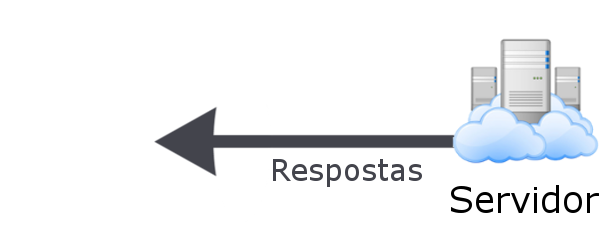


**Cliente/Servidor**

Uma arquitetura de aplicação distribuída, ou seja, na rede existem os fornecedores de recursos ou serviços a rede, que são chamados de servidores, e existem os requerentes dos recursos ou serviços, denominados clientes.

O cliente não compartilha nenhum de seus recursos com o servidor, mas no entanto ele solicita alguma função do servidor, sendo ele, o cliente, responsável por iniciar a comunicação com o servidor, enquanto o mesmo aguarda requisições de entrada.





**WEB**

Uma aplicação web é composta por dois atores principais: o cliente e o servidor. O cliente normalmente, mas não necessariamente, é um navegador como o Internet Explorer ou Firefox. O servidor é uma aplicação, na forma de um serviço, normalmente hospedado remotamente.

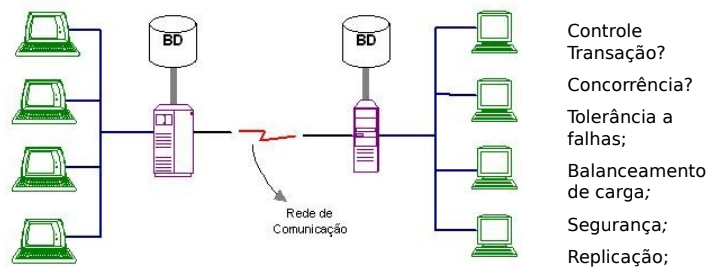
O servidor aguarda a requisição vinda do navegador, que chega através de uma requisição. Esta requisição é processada pelo servidor (+ banco de dados) e uma resposta é gerada, esta resposta e a interpreta, normalmente construindo uma página visual ao usuário.



**Distribuídos**

A arquitetura de um sistema distribuído deve levar em conta quais são os seus componentes de software e como eles interagem entre si, para que com esta análise, seja possível decidir qual é a melhor maneira de alocar as máquinas disponíveis para constituí-lo.

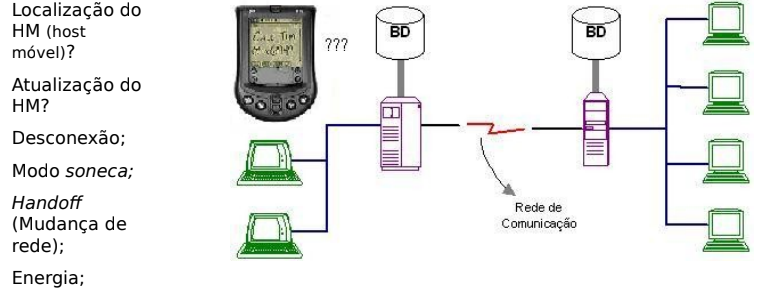
Fatores como o desempenho ou o tempo de resposta de uma certa máquina podem ser utilizados para se decidir quando criar novas instâncias daquele software ou camada, ou quando derrubá-las para economizar recursos.



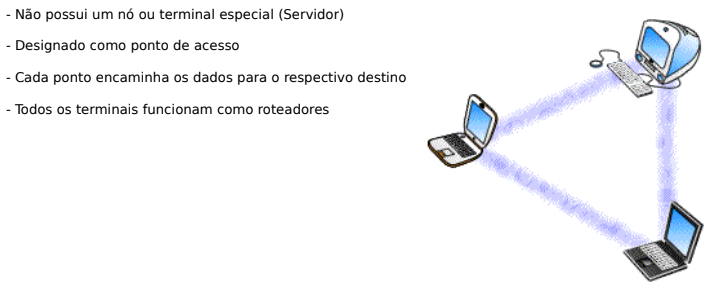
**WebService**

Serviços, assim como componentes, são considerados blocos de construção independentes, os quais coletivamente representam um ambiente de aplicação.

Qualquer interface exposta por este client service também o qualifica como um serviço a partir do qual outros serviços podem solicitar informação, eles tendem a estabelecer um sistema ponto-a-ponto, onde cada serviço pode atuar como cliente ou servidor.



**Móvel**



***Modelagem e Modelos***

As modelagens são úteis para a compreensão de problemas além de proporcionar a comunicação com qualquer um envolvido com o projeto (clientes, especialistas de domínio, designers etc).

Modelos são abstrações que retratam a essência de um problema ou estrutura complexa, filtrando detalhes não essenciais, tornando o problema mais fácil de entender.

***Diagrama Caso de Uso***

* Ator: Boneco, São suas propriedades o nome e uma descrição resumida do ator.
* Caso de Uso: Bolinha, São suas propriedades o nome do caso de uso e sua especificação.
* Relacionamento: Risquinho, são usadas para descrever os relacionamentos entre os atores e os casos de uso que eles participam.
* Simples: \_\_\_\_\_\_\_\_\_\_\_
* Extend: Caso de extensão de uso (opcional) <- - - - - - -

A seta (tracejada) de um relacionamento de inclusão recebe o estereótipo <<estende>>.

* Include: Caso obrigatório de uso - - - - - - ->

A seta (tracejada) de um relacionamento de inclusão recebe o estereótipo <<inclui>>.

* Herança: Significa que o herdeiro possui o mesmo comportamento que o ator do qual ele herda.

A seta (sólida) de um relacionamento de herança não recebe estereótipo.

**4. Considerando a constante evolução da arquitetura do software, comente sobre o funcionamento da arquitetura WEB, apontando a sua importância para os dias de hoje.**

A arquitetura da informação é importante porque organiza e dá clareza ao projeto, permitindo ao usuário utilizar o espaço virtual com eficácia e rapidez.

**19) Sobre relacionamentos de caso de uso, assinale a alternativa que NÃO apresenta um relacionamento de caso de uso.**

A)Comunicação

B)Inclusão

–C) Implementação

D)Extensão

E)Generalização

**18) Descreva os componentes do modelo de Caso de Uso.**

Ator

Um elemento de modelo que representa cada ator. São suas propriedades o nome e uma descrição resumida do ator. Veja Concept: Ator para obter mais informações.

Caso de Uso

Um elemento de modelo que representa cada caso de uso. São suas propriedades o nome do caso de uso e sua especificação. Veja Artifact: Caso de Uso e Concept: Caso de Uso para obter mais informações.

Associações

As associações são usadas para descrever os relacionamentos entre os atores e os casos de uso que eles participam. Este relacionamento é comumente conhecido como uma "associação de comunicação".

**6. Comente sobre modelagem e modelos.**

As modelagens são úteis para a compreensão de problemas além de proporcionar a comunicação com qualquer um envolvido com o projeto (clientes, especialistas de domínio, designers etc).

Modelos são abstrações que retratam a essência de um problema ou estrutura complexa, filtrando detalhes não essenciais, tornando o problema mais fácil de entender.