Client-Server Architecture

 [Name of the Writer]

[Name of the Institution]

Client-Server Architecture

A computing model in which a server caters to the needs of one or more clients is called client-server architecture2. In this type of architecture, the server holds, manage, and deliver most of the resources that a client would request. This architecture has more than one client computers connected to a centralized server through the internet. Client-server architecture is often termed as a network computing model. This is because an underlying network is used to share resources between client and server.

Client-server architecture can be termed as producer-consumer architecture in which the server is defined as producer and consumer is the client1. A server holds, manages, and furnish resources to the clients on demand. The resources can be granted access to an application or a printer, and file sharing.

The working of the client-server system is easy to understand. For instance, a user (client) wants to log in their Facebook account. The user accesses the website of Facebook. This access is the request that the user (client) sends to a server. The server, then, in response processes the request of the client and send a ‘response'. This ‘response' is the appearance of the Facebook website on the user's (client) screen. This request and response take place within a fraction of a second without us noticing the number of technicalities that goes in this process.

A client can be entertained by multiple servers at a time1. Similarly, the server can address the requests of many clients simultaneously1. The internet that we use is built on the client-server architecture.



**Client-Server Architecture**

End Notes

1. Bertocco, Matteo, Franco Ferraris, Carlo Offelli, and Marco Parvis. "A client-server architecture for distributed measurement systems." IEEE transactions on instrumentation and measurement 47, no. 5 (1998): 1143-1148.
2. Govett, Ian Robert. "Client/server architecture supporting concurrent servers within a server with a transaction manager providing server/connection decoupling." U.S. Patent 5,761,507, issued June 2, 1998.