**Assignment 2**

Research on the Client and Server model in **software** design.

* Motivation behind the client/server paradigm?

The motivation behind the client-server paradigm is to combat the rendezvous problem. The rendezvous problem is a situation where in a pair of communicating applications, communication is cumbersome. This is because the first application starts execution and sends a message to the other application(its peer). Within a few milliseconds, the first application determines that the peer does not exist yet therefore it sends an error message and exits. While the second application starts execution, it finds that the peer has already stopped execution. The probability of the applications communicating simultaneously is low even if the applications continuously retry communicating. As a result, no communication occurs.

Since TCP/IP does not respond to incoming messages on its own, the client-server model solves the rendezvous problem by ensuring that in any pair of communicating applications, one side must start execution and wait, for an indefinite amount of time, for the other side to contact it.

* Characteristics of client/server paradigm?
  + Terminology and Concepts

**Client:** an application that initiates peer-to-peer communication. Whenever a client application executes, it contacts a server, sends a request and waits for a response.

**Server:** Any program that waits for incoming communication requests from a client. Upon receiving a client’s request, the server performs the necessary computation and returns the results to the client.

**Authentication:** refers to the verification of the identification of the client by the server.

**Data security:** This is guaranteeing that data is not unintentionally revealed to unauthorized applications or compromised.

**Authorization:** This is the determination of whether a given client is allowed to access the service that the server supplies.

**Privacy:** This refers to keeping information about individuals from unauthorized access.

**Protection:** This refers to guaranteeing that network applications cannot abuse system resources.

**Standard application services:** These are services that are defined by TCP/IP and have been assigned well-known, universally recognized protocol port identifiers.

**Nonstandard application services:** Also known as locally-defined application services. These are services that are not defined by TCP/IP but by the sites.

**Remote terminal client:**  Refers to client application that uses the standard TELNET protocol for remote login.

**Electronic mail client:** Refers to a client application that uses the standard SMTP protocol to transfer electronic mail to a remote system.

**File transfer client:**  Refers to a client program that uses the standard FTP protocol to transfer files between machines.

**Web browser:** An application that uses the standard HTTP protocol to access web documents.

**Fully parameterized client:** Refers to software that allows users to specify a protocol port number as it has more input parameters compared to other software..

* Differentiate between the following concepts. Give appropriate examples.
  + Connectionless and connection oriented servers

Connectionless servers use the UDP protocol while the connection-oriented servers use TCP. While connection-oriented servers are reliable, connectionless servers are unreliable because they provide best effort delivery and do not introduce errors. With connection-oriented servers, TCP verifies that data arrives and retransmits the segments that do not arrive automatically, computes a checksum over the data to guarantee that it is not corrupted during transmission, uses sequence numbers to ensure that the data arrives in order and eliminates duplicate packets automatically. If for any reason the underlying network becomes inoperable, TCP informs both the client and the server.

With connectionless servers, the response sent back by the server to the client may be lost, duplicated , delayed or delivered out of order and the responsibility fall to server to ensure that it can detect and correct the errors.

Examples:

* + Stateless and stateful servers (File server examples - exercise 2.4, 2.5)

Stateful servers keep information about the status of ongoing interactions with clients, referred to as state information, while stateless servers do not maintain the state information. The motivation for state servers is to reduce the size of messages that the client and server exchange and to allow the server to respond to requests quickly while the motivation for stateless servers is to ensure protocol reliability by avoiding sending incorrect responses that are computed using wrong state information.

* Examples:
  + Standard client software
  + Standard Server software
  + Non standard: Software from assignment 1 (Explain purpose of software)
* Design any one of the following applications as client server programs. Describe your design. Develop a basic (Iterative server) connection oriented implementation of the application.
  + A Bus booking application that provides the following services: SelectSittingPosition(); BookBus(); Payment(); CancelBooking();