

**COMSATS University Islamabad (CUI)**

Project Report OSI Layers

for

**Gym Management System**

**(Data Communication and Computer Networks Project)**

***By***

**Saleh Arif SP19-BCS-027**

**Shehroz Ahmad SP19-BCS-032**

**BCS-5A**

***Supervisor***

**Dr. Ashfaq Farooqi**

# Description:

The purpose of Gym Management System is to manage all the records of equipment and members by using Client and Server. The transport protocol used in this application is UDP. To store data, we have used ORACLE DBMS. After establishing connection with server, the system will show the main menu from there an admin can then choose to enter in different menus which are Equipment, Members and Usage. In equipment menu, client can enter details about equipment which is then send to server to store into database. In members menu admin can view, add and search the records of members. In usage menu admin can manage details about which member uses which equipment. For further explanation, we will divide the logic of project into OSI model layers for better understanding of the whole process.

# Application Layer

In our project “Gym Management System” application layer is defined as the programming logic in the client and server sides. The “Protocol” or the functioning of the code at runtime. Different menus for entities, take input from clients, different ways it reacts to inputs from the user and gives its output can be termed as the application layer implementation in our project. All the cases in switch statements represents Application layer.

In Application layer, we also handle all functionality related to objects and instances, data insertion and retrieval, how we send this data by converting objects into bytes using serialization and how we send this data using streams.

# Transport Layer

In Gym Management System, we are using ‘UDP Protocol’ of transport layer. To make communication possible between the client and server side, we are using Datagram Socket. It opens a socket that uses the IP address of the nodes and the port number within the nodes to communicate. Port numbers are differentiated at the transport layer to communicate the correct message to the application layer/process. Thus, these sockets are implementing the transport layer in our project.

In our project for creating a UDP datagram socket we use the following statements for server and client respectively.

DatagramSocket serverSocket = new DatagramSocket(132);

DatagramSocket clientSocket = new DatagramSocket();

# Network Layer

In Network layer, we provide the Datagram packets the IP address and Port number of the node we have to communicate a packet to. IP addresses are defined at the network layer and differentiate between two nodes. This unique IPv4 or IPv6 address is used to reach the correct device we want to communicate our message to.

The client is provided the IP and port number of the server so that it can initialize communication. When the server receives the Packet sent from the client, it also saves the IP and port number of the client so that it can send a message (within a packet) back.

In our project, to send data from client to server we will create an instance of DatagramPacket with data to be send its length, IP Address of client and port number which we will use to communicate data. Then, we will push the packet into the socket. Similarly, to receive data we will create an instance of DatagramPacket in server as well.

sendPacket = new DatagramPacket(sendData, sendData.length, IPAddress, Port);  
clientSocket.send(sendPacket);

receivePacket = new DatagramPacket(receiveData, receiveData.length);  
serverSocket.receive(receivePacket);

# Link Layer

Link layer is responsible for moving the data frames from one node to another. Link layer also provides the services of error detection and correction in the frames communicated. Firstly, it uses MAC addresses to make sure the data frames are being sent to the correct recipient. It uses Checksums to see if the data is intact. Link layer encapsulates the datagrams and adds its headers to them. One of the main information in the header is the MAC address of the source and destination. It also controls the flow between the sender and receiver. The sender sends a number of frames to the other node and the receiver sends an acknowledgment after it receives the max amount of packet it can receive simultaneously.

The NIC or network cards or ethernet cables are implementing the link layer. Error checking happens at these sites. In our project, the Datagram packets are converted into frames using the header of the link layer then sent forward to the physical layer. In our project, we use the MAC Address of the sender and receiver.

# Physical Layer

The physical media used for the data to travel as signals between two nodes is the physical layer’s concern. We use periodic or non-periodic composite signals to communicate our data. The bandwidth of the signals is of concern here. The amount of data being simultaneously communicated is directly proportional to the bandwidth.

The Wifi (802.11 protocol) or the ethernet cable that connects you to the router (that gives access to the internet) will be used to communicate the data in the form of signals to the receiving node. In our project, physical layer represents medium through which data is transferred between client and server.