Design issues for the layers:

Some of the key design issues that occur in computer networks are present in several layers. Below we have briefly mentioned some of the more important ones.

1. Every layer needs a mechanism for identifying senders and receivers
2. As a consequences of having multiple destinations, some form of **addressing** is needed in order to specify a specific destination.
3. There is a need for **rules for data transfer.** In some systems, data only travel in one direction; in others, data can go both ways. The protocol must also determine how many logical channels the connection corresponds to and what their priorities are.
4. **Error control** is an important issue because physical communication circuits are not perfect. Many error detecting and error correcting codes are known, but both ends of the connection must agree on which one is being used.
5. Not all communication channels **preserve the order of messages** sent on them. To deal with a possible loss of sequencing, the protocol must make explicit provision for the receiver to allow the pieces to be reassembled properly.
6. **Fast sender swamping slow receiver** with data. The possible solution to this is **flow control** which limits the sender to an agreed-on transmission rate. The other solutions involve some kind of feedback from the receiver to the sender, either directly or indirectly about the receiver’s current situation.
7. Another problem that must be solved at several levels is the **inability of all processes to accept arbitrarily long messages.** This property leads to mechanisms for disassembling, transmitting and then reassembling messages.
8. When it is inconvenient or expensive to set up a separate connection for each pair of communication processes.
9. **Multiplexing** and **demultiplexing** is done transparently and is needed in the physical layer.
10. When there are multiple paths between source and destination, a route must be chosen. Sometimes the decision might be split over two or more layers. This terminology is termed as **routing**.