Optional:
Encryption
Pass down channel through layers

Compulsory:
Hamming code data integrity

Transport – Dom
Network – Nathan
Link – Huw

Transport: Poll device until we receive a response to say it can accept the message
Maybe queue messages if we can't see if yet??
Hamming code for data integrity
'Send last again' if data isn't integral
Break up large amounts of data into segments of fixed length
Check none of the data is corrupted else ask for retransmission

Network: Routing tables with mesh networking, work out which il mattos can see which il mattos

Link layer: Put packets into frames in correct order
Give each frame a number
Handshake and low level acknowledgement of each frame

Store everything in strings
C++

Destination = 8 chars, eg Huw, or Nathan

Application layer:
Char SendData[1000] = 0;
Char ReceivedData[1000] = 0;

Send_data(destination, &data[0]);

Flag = Receive data(&data[0]); //Receive data returns 0 if no new data
If (flag)
    Display_new data

Transport layer: //Try and use classes in each layer?
Char segment1[120]
Char segment2[120]
……
Char segment5[120] //Or max amount of segments

Char* sendpointer;

```
Char* recievepointer;

Void send_data(char destination[10], sendpointer*)
{
for (I = 0 to 120)
        segment1[i] = sendpointer[i]
for (I = 120 to 240)
        segment2[i] = sendpointer[i]
//copies all data in x number of segments

segment.hammingencode();


send_segment(destination, segment1)
send_segment(destination, segment2)…..
until segment 5

}
```