C++ PRACTICAL TEST

1. You have been hired to implement a brand new product for a company. This product is an Environment Monitoring System consisting of hardware and software. The hardware is already finished and in production, so you have been hired to develop the software components. The system should have a central server responsible for management of elements (CRUD), interaction with hardware and report generation. The system will have small servers to support groups of up to 600 sensors, collecting data and making it available to the central server. Make a brief modeling for the system, describing the system architecture and listing the software modules and the way and order you would implement these modules. Highlight what you consider to be critical points.

2. For this test, you are provided with three source files: frame.h, frame.cpp and main.cpp. There you will find the Frame Class and a main function skeleton where you must add your solution code.
   The simIncommingBytes() function in main.cpp generates jitter bytes and single-byte-frames in a pseudorandom fashion and queues them in the global buffer incommingBuffer. Frame encodes and decodes character-oriented ASCII frames. You don't need deep knowledge about how frame encoding/decoding is done because Frame offers those functions, so just read Class Documentation in source code and use it.
   You must add the code needed to main.cpp to decode frames queued in incommingBuffer and look for the occurrence of a given sequence. Your program must ask the user for the desired sequence at the beginning of the main function. Your program must finish when it finds the sequence and must print a summary of the total amount of bytes processed and the total amount of valid frames found. Avoid using more global variables and try to make your code as short and clear as possible. Don't try to write foolproof code; assume that the user will always type in correct values.

   The output of your program should be like this (input method may vary at your will):

```
# Please enter the sequence you want to look for (0-4 only): 0012234
  Sequence found after 28254 bytes y 3002 valid frames.
```