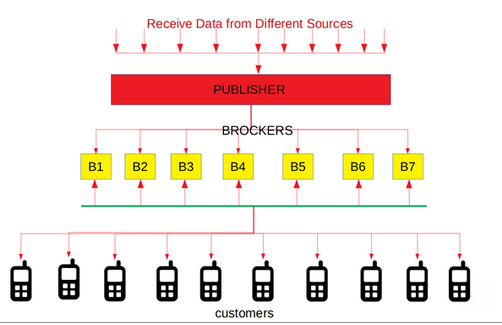
**Questions and Answers**

**1. a) Model Proposed:**

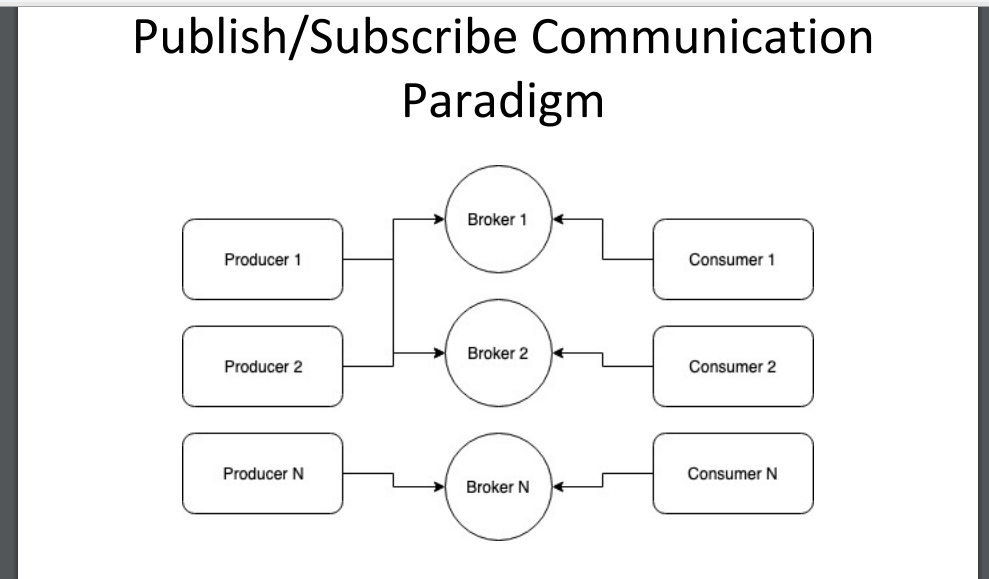
* Publisher is the main application node that will receive data from different sources and pass it on to Brokers.
* Publisher has all Broker information and instead of providing all the information received by Publisher, it will only transmit the required information to the Broker.
* Broker has the ability to handle requests from Subscribers who are the third part of the image.



* When the Subscriber sends a request to get the data, the request is automatically transferred to the Broker who is responsible for maintaining the data required for this request (The Broker gets the data from Publisher) and the Broker will provide the desired response to the Subscriber.
* Application development is divided into two parts first is the development of distributed architecture (not just analysis and design) for information management. The second part is to develop the Android application as a client, along with its UI.
* The main part of the Android app is a map showing the status of the bus on the Android device.
* The Subscriber will be able to send the filtering or a specific route request for the bus. The request will pass through the Broker who will provide the current state of the bus on the application map.

**1.b) Answer from the Client:**

A remark in relation to publishers in the diagram: The publisher will not be one, but there will be many. (Generally, we have been told by our client that they will not be asked for more than 3). Essentially they read the data from the txt file. I attach below the diagram that client gave us with publishers brokers and subscribers. Everything else from the model proposed is right.



   **2.a) Question:**

 What exactly is the third party API mentioned, Is it a specific or does it mean Publisher?

**2.b) Answer:**

It means publishers

      **3.a) Question:**

  Can you give us some examples of the keys for which this user will be responsible?

**3.b) Answer:**

Each broker has a specific ip + port and is responsible for certain keys. Specifically, we can have a txt file, as the customer told us, that will contain all the ip + port that brokers can use. In order to have an equal distribution of the topics that each broker gathers, I will check the hash (ip + port) of the broker <hash (busLineId). If so, the broker will get the topic.

**4.a) Question:**    Should we use sockets anyway? Can it be done easily through rest api?

**4.b) Answer:**

He did not say so. Essentially the first phase testing will be done in our company on 2-3 computers with ip that we will know. Just to specify that each component (broker, publisher, subscriber) must have a main that will be called from different pc at a time, and in order to synchronize, we need to use threads (this suggested at least).

**5.a) Question**:  For the development, do we need a socket server to connect to the Android application such as xampp or any other socket server?

**5.b) Answer:**

He told us to use sockets and attach the code that I sent. He told us we can copy-paste it.

  **6.a) Question:**     We checked the .txt files of DB. The question is: who will update the files and how to update the files? Let's say, in line code 1151, it's in X, Y lat Long. After 30 minutes he arrived at New Location X, Y. So how exactly will he know the current coach position Publisher?

**6.b) Answer:**

He told us that there is no need for time. So what happens is that the user chooses which bus he wants, for example, 051 and simply shows him all the 051 on the existing map.

  **7.a) Question:**     Will the application UI have only the map or something simpler to make it more realistic and quick?

**7.b) Answer:**

Only google map and a field where the user will ask for what bus he wants.

  **8.a) Question**:

 Should Firebase or Pusher be used?

**8.b) Answer:**

No.

I hope to cover most of the questions. If something is not clear, I am at your disposal to explain it in more detail, or to ask for clarification.