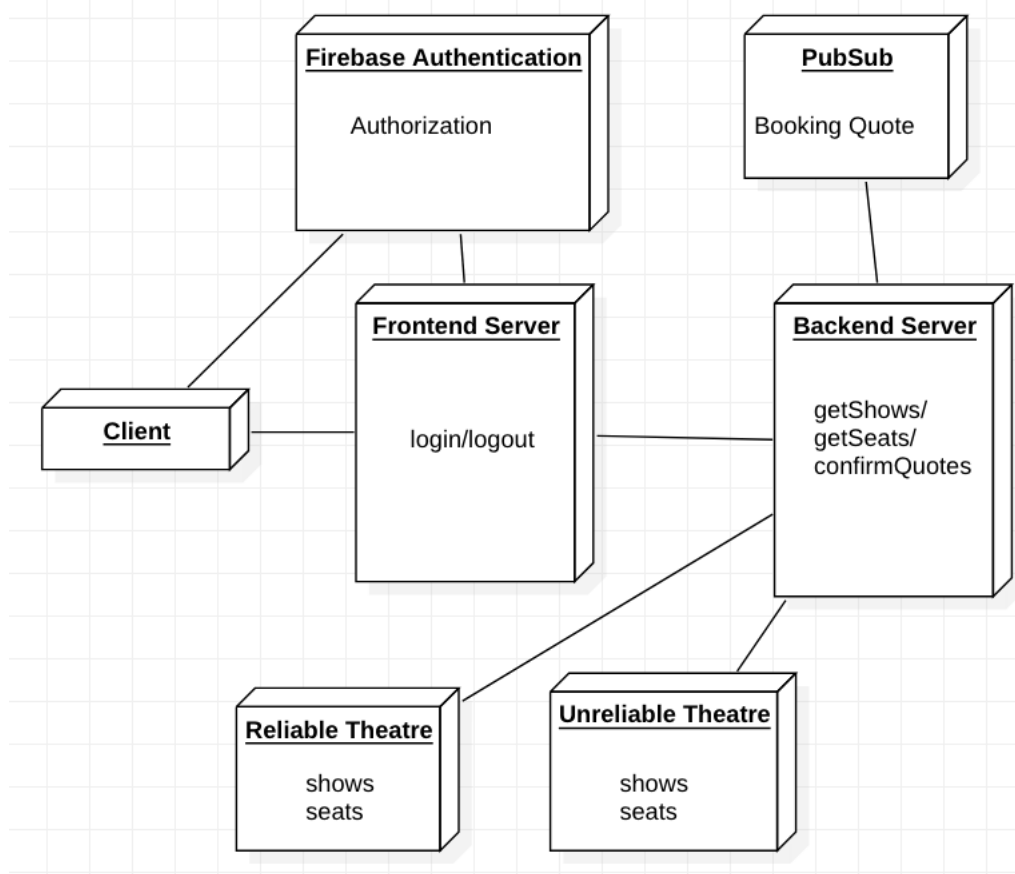# *Report:* Google Cloud

MOFAN DENG *(r0868382)*
YINQI LIANG *(r0865008)*

November 26, 2021

Question1   The backend process service about interacting with theatre to get information about shows and making bookings, pubsub holds booking quotes, frontend process service regarding authorization with Firebase Authentication.

Question 2   We use pubsub as middleware to decouple the confirming booking logic. It used to be made up of creating quotes, sending put request to company, storing the booking to database and clearing the cart. But with pubsub, we move the sending put request and storing database to another end of pubsub, which make the whole booking process as if it only contains creating quotes and clearing carts from the viewpoint of front end. This is how pubsub helps in hiding complexity. And when we use pubsub to divide the logic into two parts, we can separate the workload in a further way as long as we make an agreement on data format of pubsub message, which allows us to develop the program simultaneously and speed up.

Question 3   After collecting cart, creating quotes, we publish information required for booking to pubsub so that the /confirmCart API can response the user quickly. Pubsub holds a push delivery subscription to /subscription endpoint which call the comfirmQuotes method to communicate with company to book tickets, and then save booking information.

Question 4   When creating a booking, a quote containing company name, showID, seatID is passed. And then this information regarding making the booking will be passed as message to pubsub. It doesn't make sense to persist data and only pass references considering that in real situation different components above the pubsub might be on different server. The reference is useless. What's more, if some component breaks down, the data will vanish as well. Actually, one use of pubsub is to temporarily store the massage so that when some component recovers from breakdown, the workflow can continue without loss.

Question 5   For scalability: when we increase the number of servers, or we want to add a new financial log system to record and analyze the bookings, we do not need to make big variations on current code, because the newly added components can just subscribe the pubsub topic. For responsiveness: due to that our booking system serves as an intermediate role between user and show company. Every booking request involves creating booking, sending requests to company, saving booking information to databases. It will take relatively long time and keep users waiting. But with pubsub, the ApiController only needs to create booking and publish it to the pubsub. It can response to users quickly which represents high responsiveness.

Question 6   One of the benefits of indirect communication is to decouple the different components of the system. With pubsub, the publisher can just push message according to the agreed format regardless of whether there are more subscribers and vice versa, which significantly enhance the scalability. However, it will inevitably result in asynchronization. In other words, for those method requiring message immediately like some data required by GET method (e.g. getShows(), getSeats(), etc. in this project) to be displayed on the webpage, usability is more important. In this case, indirect communication would not be beneficial.

Question 7   No, because any successful booking of one seat will change the status of ticket to unavailable, which result in failures of following put request. And our booking data will be saved only after receiving all the successful response from the company. Once a put request fails with status code 409 (already booked), our booking method will send delete requests to roll back first and exit with an exception. The code of saving booking will not be executed.

Question 8   With role-based access control, we can simply set limitations regarding authority of roles on APIs to block illegal requests by checking token within the cookie. It doesn't have to check the manager account separately, or refer to the database whether this user have this authority.

Question 9   If switch to another authentication provider such as Shiro, the firebase authentication needs to be changed. If the authentication provider does not provide role-based access control, it would be more difficult. They should provide user identity authentication.

Question 10   Since at most time the Unreliable Theater Company only breaks down no more than 1 second, frequently though it is, we use the retry and retryWhen method of Java Webclient to deal with the 5XX error response to keep the system working at the entire time. However, If the unreliable company keeps breakdown for really long time exceeding the limitation of retry then our system will break down as well.