Assumption and consideration

The assumptions and considerations of our climate service project is listed as below:

1. **climate services are worthy our efforts**

Climate report has been an indispensable part of our daily lives. Reliable climate services management plays an important role in providing customer with accurate and useful climate services. Our project is based on the assumption that our available climate services are reliable and quite useful for both common user and scientists. The data sources of available climate services are valid and can give us accurate data and analysis.

**2. budget is enough**

We assume that our project will gain enough budget for implementation of necessary and additional functions. In real world, such a project cannot prove to be finished without enough budget support. Based on this assumption, we can continue to add additional interesting functions according to our interests.

**3. interaction between users and system is valid and secure**

Our climate services management system includes functions requiring users to interact with system or other users. “services rating” and “service recommendation” assume that users will give reasonable judgement to climate services according to their own services. Our climate services management system does not provide user with security protection and default tolerance mechanism. That is because we assume that people’s action is reasonable and no security attacks will come from outside.

**4. aim for scientific utility**

Our climate services management system is aimed to scientific utility, thus we do not provide attractive GUI and advertisement functions. Outlooking is not included in our consideration, on the opposite, we pay much attention on functionality. At the same time, our system offers APIs for social network connections, because our system is needed to be connected with other social network applications in nearly future.

**5. big data requirement**

Our implementation does not support big data applications. To support big data, our back-end database and framework need to be improved. For example, we can build our service on public cloud platform like Amazon AWS. We can use other back-end database like Hbase and Oracle database instead of MySQL. We can also configure our system in distributed running environment like Hadoop or Sparks.