1. **Architectural Styles**

As a web application, our system is based on the Client/ Server model, which benefit the system for centralized data storage, and a scalable amount of users accessing and providing data. The clients of the system are different applications run on different user machines, which allow users to login the system, upload their monitoring data, and get or give the data analysis and interactive with social network. While on the other hand, the server which runs on a separate machine is in charge of the system database and all the backstage requests process. The server can be further partitioned into several basic functional units, which in charge of data management, data transit, and user application communication.

By using this client/ server model, the system reduced the procedure done on the client side to a minimum degree, and leave all the process work to the server. In this way, we can provide a light user application, also a better coherence with all the user data. We could have a light front-end application for user to run fast on most device. But on the other hand, it requires a powful server to deal with and store lots of data effectively. The centralized architecture make the server act as both the database and the central data processor in whole system.

1. **Identifying Subsystems**

We can basically divide the system into two subsystems: the Client side and the Server side. The Client subsystem is consisted of Web Brower and other User Interfaces, such PHP, HTML and XML files. As for Server side, there is a central controller unit to controll the whole works inside server side. User authentication unit can validate user information and send user request to central controller. Then the controller forwards the request to Server Logic unit to parse and process it. After analyzing request, the central controller will connect to Database. The Database package is resposible for data storage and providing external data such as Social Network information.

