System Design Progress

Step 1: Requirements clarification

- **Functional Requirements**
  - **Read** and **Write** function
- **Non-Functional Requirements**
  - Consistency and Availability
  - Accepted latency
- **Traffic**
  - Active user
  - **Read**/**Write** request per sec

Step 2: Design System function by function

- **One function one diagram**
  - Ex. Read function using one diagram, write function using another diagram
  - Design Data Schema for this function needed.
  - Design API interface for this function (input and output)
- **Common System components**
  - **Load balancer:** Avoid single point failure
  - **Cache**: Improve the read performance.
  - **Message queue:** Improve the write performance.
  - **Transaction:** Ensure the data **consistency** during writing, using SQL.
  - **NoSQL:** Good for large data partition

Scale estimation (Optional)

Raw Data

- **User**
  - number of active users
  - Average session time of active users
  - Write/read requests per users.
- **Requests**
  - Read: Read requests per second, Size of Read
  - Write: Write requests per second, Size of Write

Requests per second estimation

$$\text{Request per second} = \frac{\text{Requests per users}}{\text{Average session time}} \times \text{Peak active users}$$

Bandwidth estimates

- $\text{request bandwidth} = \text{Request per second} \times \text{size of request}$

Storage estimates

- Data size for one day.
- Data size for five years.
- **Memory estimates:** 20% of read request per day