System Design Flow

1. Understand the problem and scope:

- define the use cases, with interviewer's help
- suggest additional features
- remove items that interviewer deems out of scope
- assume high availability is required, add as a use case

2. Think about constraints:

- ask how many requests per month
- ask how many requests per second (they may volunteer it or make you do the math)
- estimate reads vs. writes percentage
- keep 80/20 rule in mind when estimating
- how much data written per second
- total storage required over 5 years
- how much data read per second

3. Abstract design:

- layers (service, data, caching)
- infrastructure: load balancing, messaging
- rough overview of any key algorithm that drives the service
- consider bottlenecks and determine solutions