Be absolutely sure you understand the problem being asked, clarify on the onset rather than assuming anything

b) Use-cases. This is critical, you MUST know what is the system going to be used for, what is the scale it is going to be used for. Also constraints like requests per second, requests types, data written per second, data read per second.

c) Solve the problem for a very small set, say 100 users. This will broadly help you figure out the data structures, components, abstract design of the overall model.

d) Write down the various components figured out so far and how will they interact with each other.

e) As a rule of thumb remember at least these:

1.processing and servers

2.storage

3.caching

4.concurrency and communication

5.security

6.load balancing and proxy

7.CDN

8. Monetization: if relevant, how will you monetize? eg . What kind of DB (will mysql do ? or nosql fits btr? ), do you need caching (almost always !) and how much, is security a prime concern?

f) Special cases for the question asked. Eg say designing a system for storing thumbnails, will a file system suffice? What if you have to scale for facebook or google? Will a nosql based db work?

g) After I have my components in place, what I generally try to do is look for minor optimization in various places according to the use cases, various tradeoffs that will help in better scaling in 99% cases.

h) Scaling out or up

i) Check with the interviewer is there any other special case he is looking to solve? Also it really helps if you know about the company you are interviewing with, what its architecture is, what will the interviewer have more interest in based on the company and what he works on?

know about the companies work: Go through the engineering blog of the company you are interviewing in (or if it is a startup go through the link of the company closest to yours)

Stage 4: common designs questions Try attempting the FAQs on design using the template. "Practice, whiteboard, mocks, revice"

Design Twitter: <http://blog.gainlo.co/index.php/2016/02/17/system-design-interview-question-how-to-design-twitter-part-1/>

References:

<https://github.com/checkcheckzz/system-design-interview/blob/master/README.md>

<https://github.com/shashank88/system_design>

<https://github.com/filipegoncalves/interview-questions/tree/master/systems_design>

<https://github.com/andreis/interview>

<http://blog.gainlo.co/index.php/2015/10/22/8-things-you-need-to-know-before-system-design-interviews/>

[http://superherojs.com/#browser](http://superherojs.com/" \l "browser)