1. What Type is the problem?

1. System Architecture or Object Oriented Model or Particular Algorithmic solution?

2. IAP - What is the input? What is the process? What is the output?

2. Use Cases (Ask lot of questions)

1. Shortening, Redirection, Custom, Automatic Expiry, Analytics,

2. UI and/API? Question to ask

3. Idempotency and Immutability

4. Finalize the Scope (Very important)

3. Constraints (Data storage, Traffic) (Ask lot of questions)

1. Maths

1. Bigdata statistics:

1. Facebook

2. Twitter

3. LinkedIn

4. Amazon

5. Dropbox

6. Google

2. Traffic:

1. Number of Requests per month

2. Number of Requests per sec for each type (possibly connect to use case/services)

3. Data storage

1. Number of bytes required for each types (possibly connect to use case/services)

2. Storage for 5 yrs (possibly connect to use case/services)

2. Confirm if the assumption is right

4. Abstract Design

1. List out an architecture

1. All components

1. UI

1. Small design about UI

2. Network layer

3. Concurrency (possible iOS components)

2. Services

1. Application Layer

2. Data storage Layer

3. Brief about some implementation or basic algorithm/logic

1. Hashing, Datastructure, Algorithm to use etc.

2. Confirm with Interviewer if I am in the right direction and he is happy (VERY IMPORTANT)

5. Bottlenecks

1. Check against the Maths (Constraints)

2. Traffic bottle necks

1. See if we should implement load balancer.

3. Data bottle necks

1. See if we should implement caching in-memory etc.

4. Availability bottle necks

1. Vertical Scaling or Horizontal Scaling (more in scaling)

5. CAP Theorem

6. Scalability

1. Knowledge - Tradeoff - talk!

1. Load balancer

2. DB scaling

3. Vertical vs Horizontal scaling

4. Datacenter to UI

5. DB sharding

6. Caching