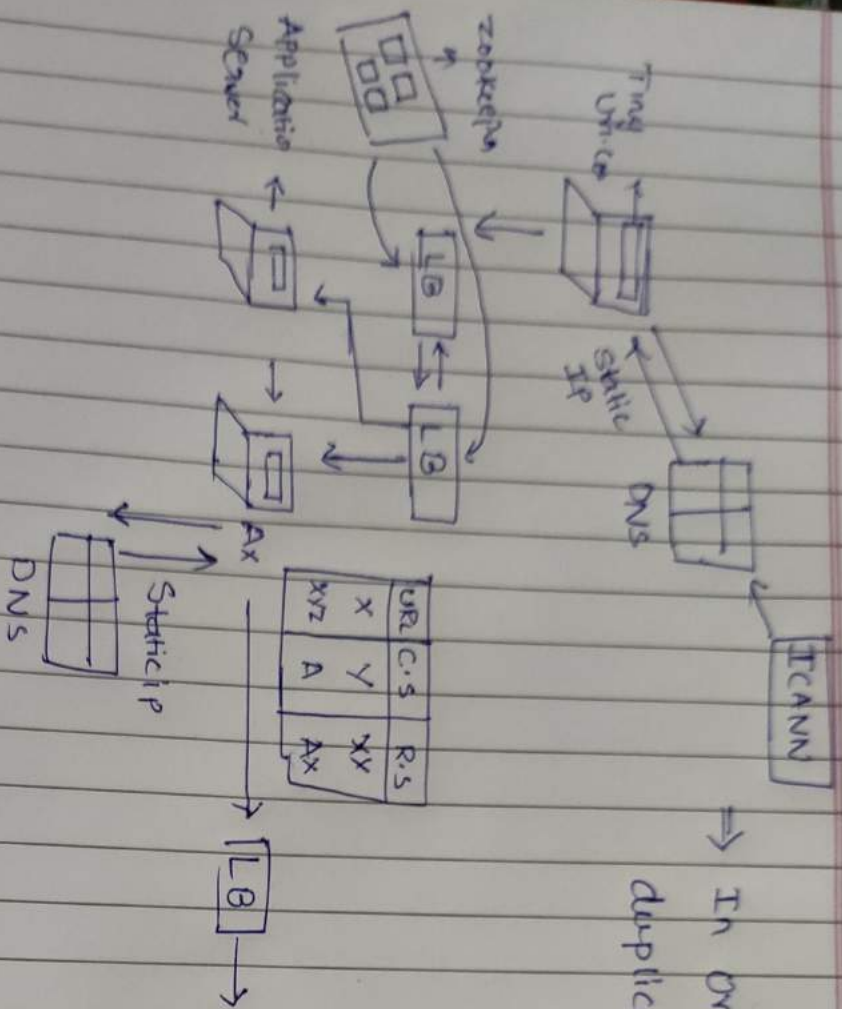


TN/URL ARCHITECTURE



⇒ In Order to reduce duplicate entry make IP URL as Unique Id

Application servers of Respective Site

→ How does your system ensure that 2 URLs never map to the same shortened URL?

Ans:- We can use hash function to Actual URL as Input and convert it into new Shortened URL

→ How will you ensure the system has very low latency?

Ans:- by Scaling Application horizontally depending upon incoming traffic and using consistent Hashing technique in load balancers in order to distribute load to application server and as well as for data sharding.

→ What will happen if the machine storing the URL mapping dies?

Ans:- We need to have a hot copy sync with actual DB as backup and to avoid Data Center failure spread DB across different regions

Date _____
Page _____

→ How do you make sure your system is consistent?
This is to say, if I have shortened a URL given the shortened URL, my system should always return the original URL no matter when I call your system?

Ans:- I will be using hash function to generate shortened URL of actual URL so the consistency should be maintained by reducing the factor of hash collision

→ How do you make sure that your service never goes down? (No outage). you have to assume that your machine will die. How do you make sure your service is unaffected by those incidents (unless all of your machines die at once - let's assume that never happens)

Ans:- I will make sure have backup to reduce single point of failure by using horizontal scaling.

I will be checking health check the instance which are working. In case if instance goes will add more system to share load and keep the things go on.