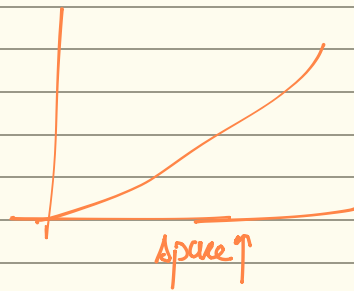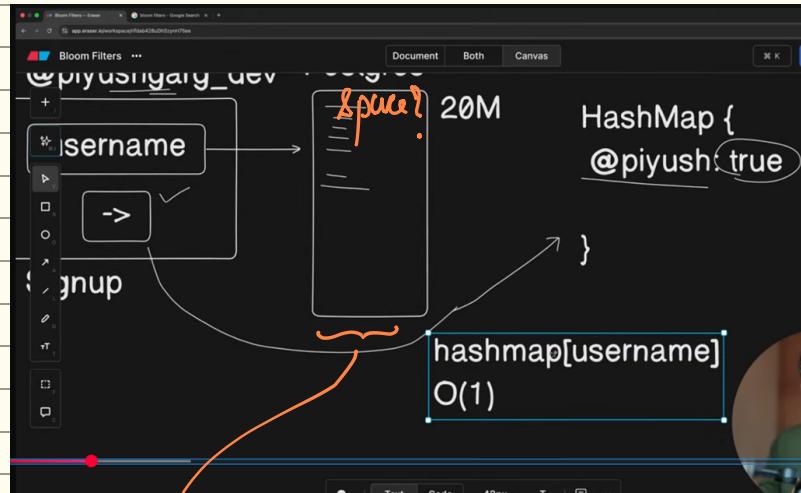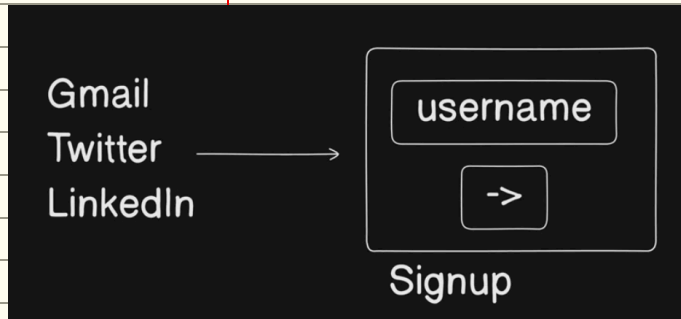# SYSTEEM DESIGN
↳ Bloom Filters

→ **BLOOM FILTER:-** Bloom Filters are a probabilistic data structure used in system design to efficiently check whether an element might be present in a set or is definitely not present.

Gmail
Twitter
LinkedIn
→ username
→
Signup

Space? 20M

HashMap {
@piyush: true
}

hashmap[username]
O(1)

↳ Lookup time in Hashmaps is O(1).

space ↑

Bloom Filters → False positive.

Hash Functions
↓

© pratyush
↳ Hash function → 01542

Buckets

	1		1		1		
0	1	2	3	4	5	6	7

ii) @pratyush ~ h - 0 2 4 8

Check if @pratyush is available? → h → 0248.

the fn check if 0248
is taken or not in the hash.
If it is take (1's in hash)
then it returns TAKEN.

@john $\rightarrow$ h $\rightarrow$ 1468 $\rightarrow$ ✗

@lemon $\rightarrow$ h $\rightarrow$ 2347 $\rightarrow$ ✓

@apple $\rightarrow$ h $\rightarrow$ 1268. $\rightarrow$ ✗ false positive.

∵ As the bucket size is so less, ∴ the collision increase and hence even if apple was not there the Hash function returned it a as total.

BUCKET SIZE: 9999

A Bloom filter is a space-efficient probabilistic data structure that uses a bit array and multiple hash functions to test if an element is a member of a set. It offers fast and memory-efficient set membership testing, providing no false negatives (it will always correctly say an element is not present if it isn't) but allowing for false positives (it might incorrectly say an element is present).