

Instagram Stories System Design

Functional Requirements

- ① User should be able to post a story - text/photo/video
- ② All user followers should be able to see story in timeline.
- ③ Story Publisher should know who all viewed story
- ④ Retention Period for stories - 24 hrs

P1 requirements (not in scope)

- ① Notification of new story published.
- ② Move story to archived storage after 24 hrs
- ③ Reply to story (chat)
- ④ Track progress of stories

Non Functional Requirements

- ① Highly available
- ② Less latency (max 10 sec) - follower should be able to see story as soon as it is posted.
- ③ Storage - Eventual consistency is fine
- ④ Durability - for 24 hrs unless deleted by user.

Capacity Estimations

1.3 Billion Active users

1/100 users post story - 10 million users

Avg - 10 stories/user/day

5 photos | 3 Text | 2 video

traffic - 10M user * 10 stories per user
= 100,000,000 stories / day

Post story = 1150 requests / sec
TPS

view story TPS
avg 100 followers / user

avg 1/10 view story posted = 10

TPS = $1150 * 10 = 12 \text{ k request / second}$

Storage Estimations :

10M users
10 stories - 5 photos | 3 text | 2 video
1MB 1KB 10MB

$\sim 12 \text{ MB / user}$

$10M * 12 \text{ MB} = \underline{120 \text{ TB}} / \text{day}$

Story will be auto deleted after 24 hrs

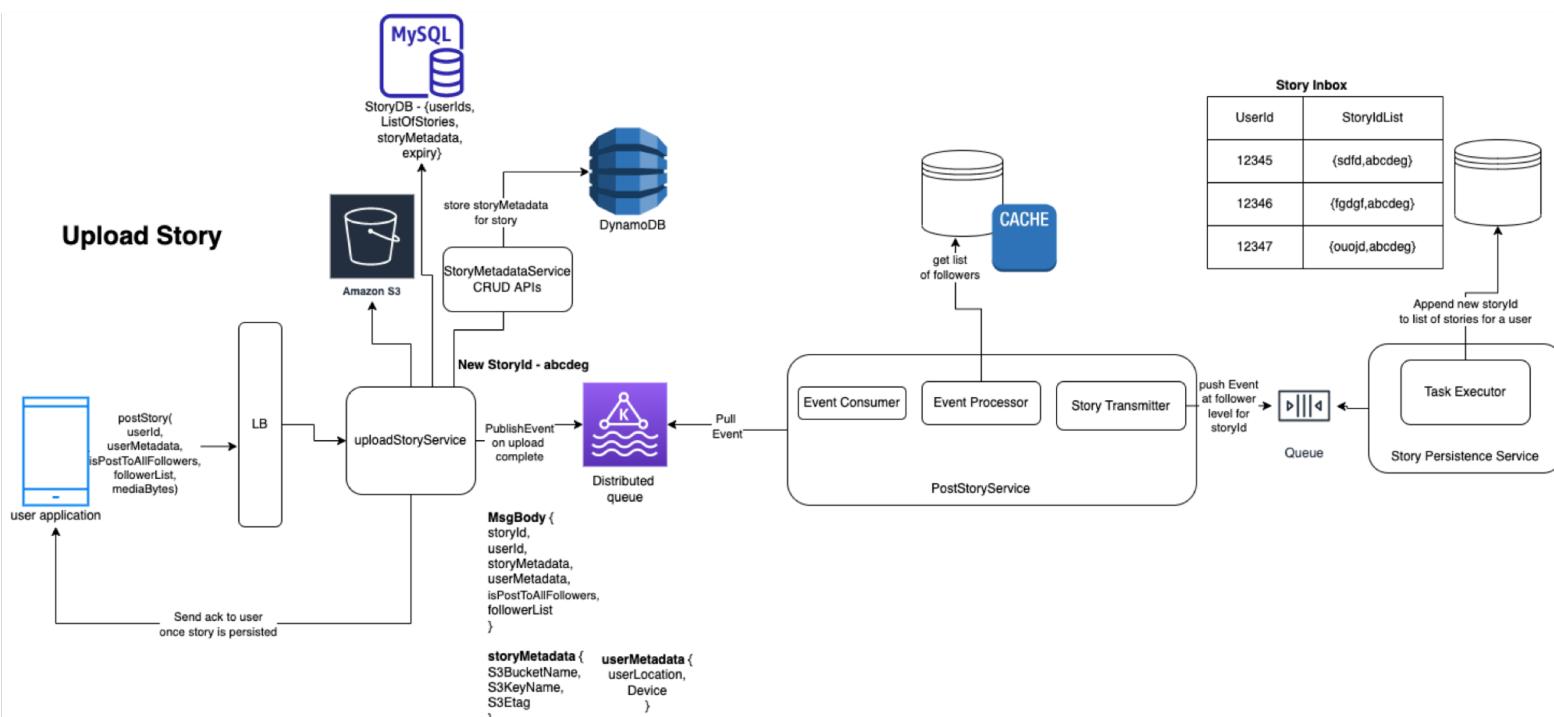
→ Extra storage to maintain story metadata,
who all viewed the story OR any other
analytics.

High Level System Design:

System will comprise of -

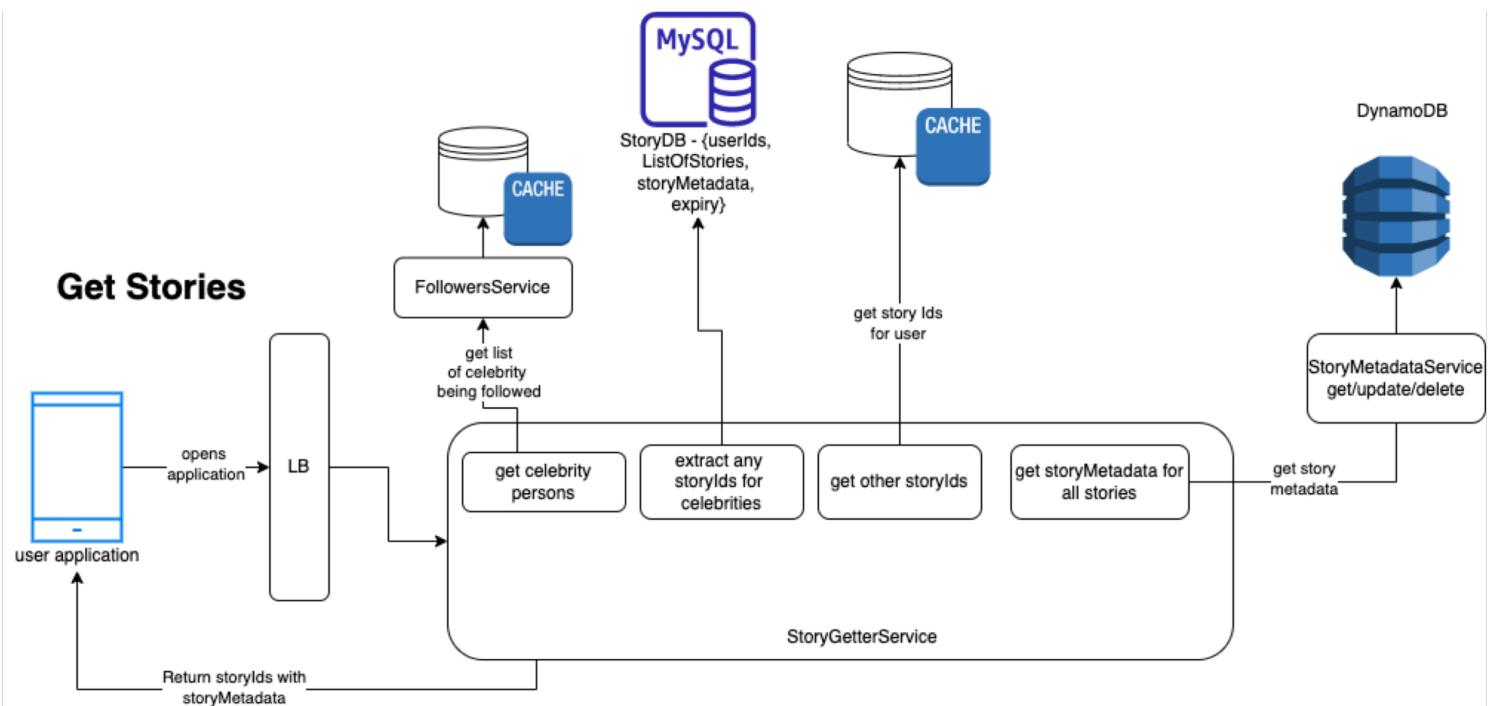
- ① Upload Story System
- ② Read Story System

[A] How will the upload System looks like?



- Make system event based.
- Asyn workflow - user is sent response as soon as story is uploaded to S3 Bucket.
- If single DB is accessed by multiple systems, create a separate micro service for DB CRUD operations.
- Create a Failure Handling in system - Failure at any step should not require complete work to be done again. It should re-run from place it failed.

B Get Story Feed for a user



C Read Story

