

## Problem Statement: Design a live video broadcasting system (Like **ESPN** or **Hotstar**).

- **Service Definition:**

- Requirements: TV shows, Movies, Sports content
- Most important feature is Live broadcasting
- Firstly use the User's online time or live stream watch time and content type and save it for use of data analysis for recommendation's page.
- Smart recommendation service which uses the users live and content watched data to suggest more shows or similar content.
- Notification Service so whenever a **subscribed show or movie or IPL match or Premier League** starts the User should get notified.
- Watch Time API → Using the total time streamed and the content watched information and its type I could be able to map it with other users data and find the most viewed content of this week and also use it for **More for You or You may like this page**.
  - **Notification API** → I could use the notification service similar to that of what **You-tube** follows that whenever popular show or movie or match goes live start to notify by technique like in 1 min send it to 10000 users and next minute to next 10000 and cover all the Users.

- **Engineering Requirements:**

- The Servers will be definitely critical in serving or handling over a million requests simultaneously, there could be a situation when there are too much requests and can cause the server to crash eventually
  - The better way to handle it is of course a back-up server but instead also I could implement or **be ready in advance**

- like for e.g. If I know that there is a Primeir league or an IPL match of qualifiers then in advance I could prepare or be ready if I usually need 4 Servers then tomorrow or for tomorrow add 1 Server to take care off.
- A good hash function needs to be implemented for use of all servers like not only a Single server should get all the traffic
- Secondly the partition of Services of movies or TV shows which are locally uploaded every week or everyday at a fixed time are kept Separate with a DB which my Service will always query if User asks for TV show or Movie we already have it stored and ready to be delivered state.
  - Live shows are specially handled which is completely separate space so they do not get interlocked.
  - Also proper and safely use Sharding concepts.
- Chat and a video call is one good thing in between of the live show like in hotstar the user could be able to chat with other live active Viewers a simple chat with live emoji feature is good idea to add a video call is little tedious to handle as it needs some extra resources and also video call quality might get affected sometimes
  - Add a viewer as a friend to chat with him
  - chat is great feature to add here with help of existing services and apis above
- A user gets notification on registered email or account then on click of notification for live show he is directed to the page with live show is rendering and after completion of the event or after it gets end ask User for feedback and then using the watched data, make analysis for future recommendations.