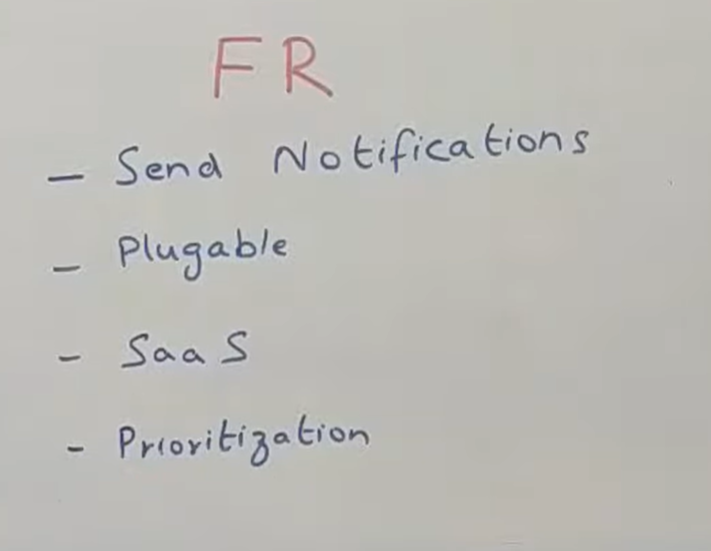
                                                                                 2.5 Notification system at scale

**Requirement :**

How can we design Notification system that is scalable enough. Let say we are building e commerce application or booking system then will always have notification service which will use notify to consumer. Now let look how can we build a notification service

**Lets look at some functional and non functional requirement**



**It shou**ld be able to**send notification**

It should be **pluggable**means let say we want to support sms an email as one form of notification. Now tomorrow someone say I want to support in app notification as well so it should be easy enough to add that. It can be further extended to lot of notification . Example : whatsapp notification

It should be built as **SaaS**product now why saas we should now who is sending what number of notification

**What is rate limit examples?**

For example, **a server can have a rate limiting component that implements a fixed window algorithm that only accepts 100 requests per minute**. The time-frame is fixed, and it starts at a specific time. For example, the server will only serve 100 requests between 10:00 am, and 10:01 am. At 10:01 am, the window reset

Notification system should be able to overall put a **rate limit**  across all the users.

Across all the platforms saying a particular user should not get more than five notification in a day.**example**If we give to amazon then there are multiple business internally  and if all of sending notification then it is very bad user experience

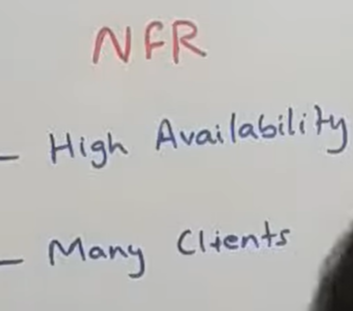
There could be certain amount of classification done. So there are two kind of notification.

One is **transactional notification**saying u have made an order this much amount of money has been deducted from your account . Transactional notification are ok to get but **promotional notification** should always have a rate limit

If we giving externally to other company as a product then you might want to put a rate limit saying that how many request can you make for this server.

The next thing is **prioritization will** support multiple priorities of messages and certain message are high priority and some are low. Let say **sending OTP**then it is very high priority. But for promotional message even if it delay half an hour doesn't matter.

We need to process high priority message first then low priority message.



**NFR**

**Software as a service**

What is SaaS? **Software as a service** (or SaaS) is a way of delivering applications over the Internet—as a service. Instead of installing and maintaining software, you simply access it via the Internet, freeing yourself from complex software and hardware management.

**What SaaS is used for?**

Software as a service (SaaS) **allows users to connect to and use cloud-based apps over the Internet**. Common examples are email, calendaring, and office tools (such as Microsoft Office 365). SaaS provides a complete software solution that you purchase on a pay-as-you-go basis from a cloud service provider.

This platform should always be available why  let say because if u r planning to build it a saas product which can be used by other companies then downtime would really cost a lot .

It should be in easy enough to add more client . We should have attribute saying how many clients have made how many number of request.

**Lets look at overall architecture of whole system**

**If** we are building small use case let say want to send email notification to some customer given some criteria we can build all of this as one deployable service and put all the logic in one place.

But if u truly want to build as a saas product wherein enormous number of client woud be using it for lot of  notification then this type of system would probably be worth it .

**Let look at starting point**

**The** starting point of whole system is couple of client i.e client 1 and client 2 could be any number of client who want to send an notification . Now there are two or **three kind of request** that they can send you . **But all of those reques**t would come into something  called as a **notification service** which is an interface for us to talk to the other team in the company , other companies or anything that we want .

**There are two kind of request mainly.**

1-One is where they tell you that i want to send this particular content to this particular user, let say a email id kind of thing as an email

2-Other request could be saying i have this user id and send  them this notification and you decide how you want to send whether as an email, as sms or whatever .

Normally the first kind of model would be use by other company where they want to decide  that they need to send as an sms or email. The second model would generally be used when you are building this to be consumed  within your own company.

But normally as saas product its good to have both the interfaces.

The idea of**notification servic**e is for most kind of request  it will take the request put it into kafka and respond back to the client saying i have taken the request and i will send notification

In couple of second at max. The n/f service would do very basic validation saying that email id and user id should not be null. All of those validation would happen in component n**otification validator and prioritizer .** it does a couple of thing . **validation** is one part of it . The main thing it does it decide a priority for a message so based on some attribute within the message let say will keep a message type identifier kind of a thing in a request and based on that it will decide the priority of message. Normally otp high priority second transaction message saying order has been placed. Least priority would be promotional message.so now it decide the priority  and put the event into kafka topic specific for each priority. So there will be topic for high priority message there will be topic for medium priority message there will be diff topic for low priority and while consuming the consumer first consume the high priority message  and then medium and low priority. Now come something as**rate limiter. These** two components can be interchanged

**Rate limit  does** two kind of rate limit

1- it check the client who calling me  is that client allowed to call me these many number of times.

The second thing it does the user whom i m sending this n/f am i supposed to send this n/f  to this user these many number of times.