Written Test

1. In order to allow “watch later”, the data we need is the video id. We also have to make sure the id is unique. The ID will be saved to database or redux (depends on the situation). Then whenever the user want to view the list, the saved id list can be joint by the full video list for retrieving the other info purposes.
2. There are several aspects we can think of user experiences, region and resources. We can use a cache server for capturing the cache of the images (use for video posters) and videos for fast loading. If the service demand is high, we can also scale up our server by adding a ngnix server for load balancer pointing to a lesser loading POD (replica servers on OpenShift). Scaling up the resources like adding memory or CPU can also benefit the efficiency if the resources are highly in use (can tell by the system usage). We can also set different replica servers in different regions, depend if the service’s localization.
3. There are serval thing I am still uncertain of. For example, I am unfamiliar with the company standards, logs, styling, practices, theme, copyrights. More importantly, there should be UAT signoff and approval or even implementation approvals obtained before going to live. Also, the movie list is only for testing purposes, we will also need a real live scenario to make sure the live is correct.
4. For features to be success, we need to write a test plan for checking the service, UT, SIT, UAT test plan. The testing will be mainly focus on the addon feature, but none the less don’t miss out the regression testing. Also, we can consider adding up to dates feature by referring to popular video on demand services, like Netflix and Youtube, to see if there any useful and nice to have feature that we have missed out.