

## Tracking interactions:

If we assume that all users self-report immediately after showing symptoms, the date of infection could be estimated as 7 days before. Therefore, we inquire in our “self-check” module how long it has been since the symptoms, and represent this information using “L”.

Hence, If a user self-reports at day D, he or she might be infected initially at day  $D-(L+7)$ , the infection could start after day  $D-(L+7)+3$ . The period of interests would be between day  $D-L-4$  and day D, and all the interaction data will be requested and stored in our database for future use (if the suspected case turned out confirmed, say).

## Delivering notifications:

For a list of tokenIDs(protected UserIDs) we use hash to map it with their existing status, default to green(healthy). (Hash is a fast mapping method.)

A notification takes place when:

- (1) Someone suspected himself or herself infected and self-reported, and the tracing algorithm located the target individuals. The level of notification alert should be amber.
- (2) A medical professional confirmed infection of a suspected user, and the tracing algorithm located the target individuals. The level of notification alert should be red.

Hence, we encode the message to binaries, using separate bits for UserID, regions etc. 1 bit will be indicating the level of alert, and hash is used to capture the user's status with reduced latency. The message is put onto notification channel buffer with smart pipeline decider.

In the case that the alert bit is 1(red alert), someone has been confirmed, and this is the highest level of alert possible which must be handled with priority. Thus:

- (1) If hash(ID) is green: put the message into the notification channel buffer with very high priority.
- (2) If hash(ID) is amber: the user has been notified due to a suspected report, put the message into the notification channel buffer with very high priority.
- (3) If hash(ID) is red: the user has already been notified due to a confirmed report, put the message into the notification channel buffer with low priority.

In the case that the alert bit is 0(amber alert), someone has been suspected:

- (1) If hash(ID) is green: put the message into the notification channel buffer with medium-high priority.
- (2) If hash(ID) is amber: the user has been notified due to a suspected report, put the message into the notification channel buffer with low priority.
- (3) If hash(ID) is red: the user has already been notified due to a confirmed report, put the message into the notification channel buffer with very low priority.

The smart pipeline decider will also take consideration of multiple details (region, health condition, etc) of the user to set an appropriate priority for each message. For example, regions with known burst of viruses are more prioritized and occupy more channel resources.