Progress from 19 to 23 April 2021

* Last week all the work for trending\_user, trending\_reviews, popular\_users, popular\_reviews, top\_reviews based on location were completed in the local system, this week all the scripts were pushed to the staging server, the packages were installed, and various scenarios checked for the RestAPI results for the above urls.
* Below are the RestAPI urls which can be consumed, here if the results are required for categoryid, then enter the categoryid, and if the entered categoryid is not present in the database, then it will give a message that “this categoryid has no results”.
  + Popular user: <https://py1.realreviews.org/popular-review?categoryid=>
  + Popular review: <https://py1.realreviews.org/popular-review?categoryid=>
  + Trending user: <https://py1.realreviews.org/popular-review?categoryid=>
  + Trending review: <https://py1.realreviews.org/popular-review?categoryid=>
* For the top reviews related to product\_id and services\_id were completed in the local system, and had implemented in the staging server, however currently in the “products” table there are 25 products, but only one product has a list of “reviews” and all other products do not have reviews, hence it was not working in staging server, also for “services” there are 7 ids, and none of the Id’s have the reviews array, Ayush has clarified that it was a recent update in the API, hence old data does not have the required “reviews” array. Hence currently have commented the script with top reviews related to product\_id and services\_id, however this can be employed once we have the data.
* For recommendation, since there is no much data, am trying various combinations such number of likes and “ratings” in the reviews table as a guage, so that the recommendation will be different from the trending and popular ones, this is still work in progress, next week would want to complete, however this will not be a great algorithm due to scarcity of data, but this can be a starter.