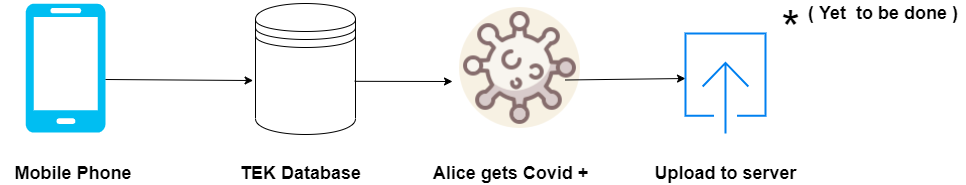
**Blog 7**

My part of the implementation was mainly focused on the storage of TEKs and received RPIs , which are transmitted from other phones. I worked on creating a SQLite database which stores TEKs as soon as it is generated along with the timestamp of its creation. This timestamp is really important as the database gets updated periodically and only the last 14 days TEKs are stored in the phone. 14 days has been generally regarded as the risk period and if the contact is older than that, then it can be overlooked.

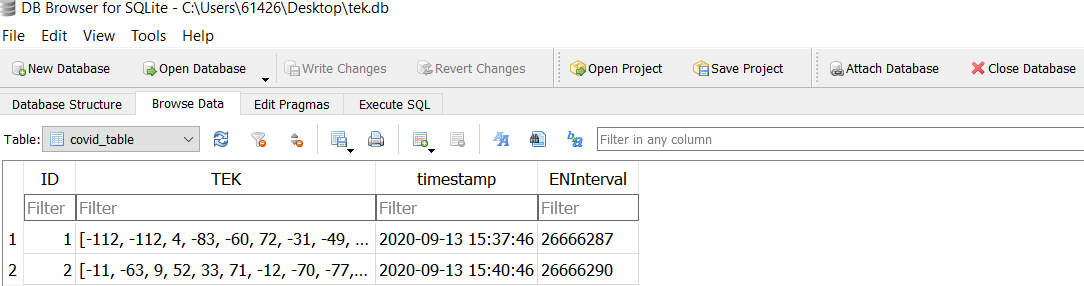
TEK database also stores corresponding ENInterval which is a unique number for each 10 minute window shared by the devices. TEKs and the first ENInterval number are sufficient to regenerate all RPIs in that 24 hour period.

Second local database stores the received RPIs from other phones Bluetooth payload and they are in a form of unique 16 byte UUIDs and are broadcasted by nearby phones

This diagram shows the use of storing TEKs



As you can see Alice is a user of the app, her phone stores TEKs daily along with the first ENinterval in the local database. If Alice is diagnosed with Coronavirus, she can give her consent to upload this database to the key diagnosis server. Users who are not diagnosed with coronavirus do not have to upload their keys at all. Below you can see the schema of the table which has 4 fields **ID,TEK, timestamp, ENInterval.**

****

This diagram shows that the received RPIs are stored locally in the database of phones of other users , so that they can be compared with regenerated RPIs , made using downloaded TEK database. If there is a match, user was in contact with alice. Comparison part is still to be implemented.

