**Tubbr Test**

**Tasks:**

1. Create an API with a single end point for creation of an event.  
   Example json : {"noun": "bill", "ts": ts, "user\_id": user\_id, "verb": "pay", "lat\_long": "192.11, 14.22", 'time\_spent\_on\_screen': seconds, 'properties': {"bank": "icici", "merchantid": "212", "mode": "netbank", "value": 200 } }
2. Trigger push notification on very first bill pay event for the user.
3. Alert user if 5 or more bill pay events of total value >= 20000 happend withing 5 minutes time window.
4. Alert TUBBR operator if bill paid , but did not give feedback with 15 mins of the bill pay event(bill pay and fdbk posted are 2 different events)

**Setup:**

* **Libraries used:**
  + Django and Django Rest Framework
  + Celery
  + Redis
  + MySQL
* **Install Dependencies:**
  + IDE Pycharm Pro
  + Run ‘`pip install -r requirements.txt’` to install the remaining necessary packages.

**Usage Module Example:**

1. MySQL
   1. Create a database with specificiation:
      1. Database name :- ‘tubbr\_test’
      2. User :- “root”
      3. Password :- “12345”
      4. Host :- “localhost”
      5. Port :- “3306”
   2. If any changes made then update the change in TubbrTest/TubbrTest/settings.py file.
2. Redis
   1. You need a Redis server as a celery broker. Download and install redis by following instructions given in “https://redis.io/topics/quickstart” link.
   2. Start the redis server on local host with default port 6379. (Terminal 2)
3. Project
   1. After setting the virtual environment , run ‘`pip install -r requirements.txt’`. (Terminal1)
   2. Run ‘`python mange.py makemigrations’` and ‘`python manage.py migrate’`.
   3. Create a super user with command ‘`python manage.py createsuperuser’` for accessing django admin
   4. After creating user start the django server by ‘`python manage.py runserver’`
   5. Open the browser and go to “https:/127.0.0.1:8000/admin/” and create some tubbr users for trial
4. Celery
   1. Start the celery in venv terminal(Terminal 3) with command ‘`celery -A TubbrTest worker -l info’`
   2. You can see the celery started and if any task is added or being executed in this terminal.
5. Testing
   1. The link for posting an event is “**http://127.0.0.1/add-event/”**
   2. The request to be made are in **api\_test.py** file.
   3. You can run **bill\_pay()** function or **fdbk\_post()** function for creating an event by making an api call.
   4. The functions will just ask for user id, the user ids are incremental number from 0.  
      So when you create the first user, the id would be 0 , second would be 1 and so on.
   5. If you want post a request from other devices just change the line 28 in settings.py to “ALLOWED\_HOSTS = [‘\*’]”
   6. The notifications and alert email function are added in **models.py, views.py, base\_functions.py**