

Evidencias

Runing Endpoint using Flask:

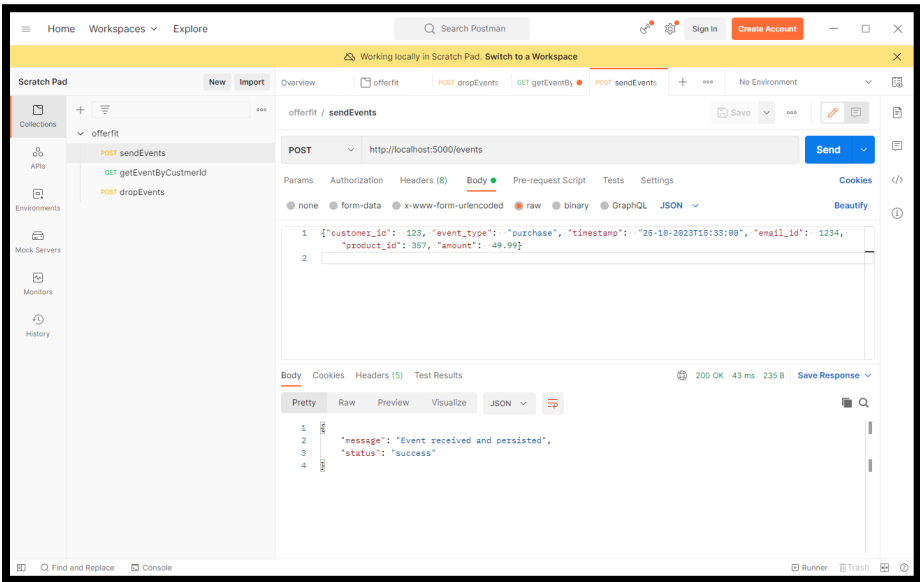
The image shows a PyCharm IDE with a Django project named 'offeritis'. The main editor displays the `offeritis/api.py` file, which defines a REST API endpoint for deleting events. The code includes the following components:

- Models:** `Event` (with fields `customer_id`, `start_date`, `end_date`), `EventLog` (with field `event_log_type`), and `EventLogEntry` (with field `event_log_entry_type`).
- Database Migration:** A `migrations` module with a `0001_initial` migration file.
- API Endpoint:** A `delete_event` endpoint that takes a `customer_id` and `start_date` as parameters and returns a 200 status code.
- Testing:** A `test_delete_event` function that tests the `delete_event` endpoint.

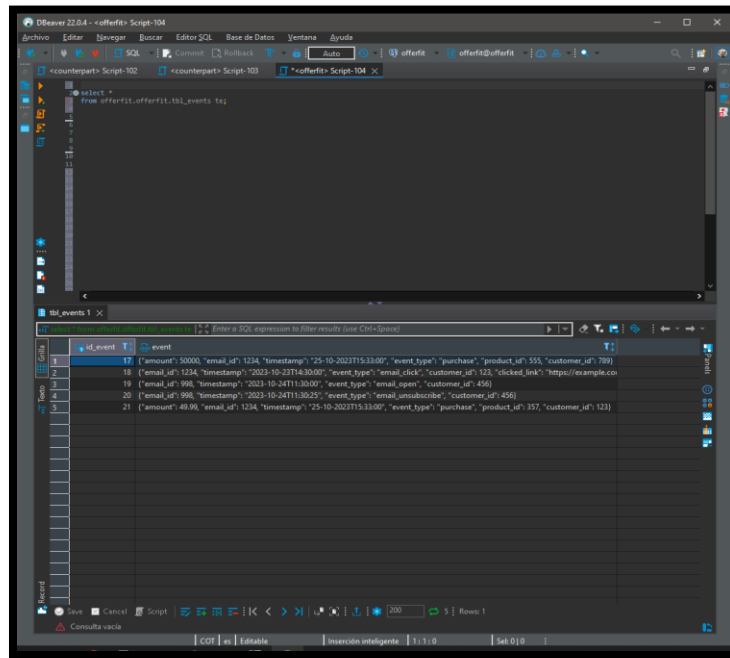
The interface also shows a terminal window with the following output:

```
PS C:\Users\user\Documents\offeritis\03 - python_scripts> python -u offeritis/api.py
table -> offeritis.tbl_events
Saving flask app "offeritis.endpoint"
Done! mount on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
Press CTRL-C to quit
* Running on http://127.0.0.1:5000
```

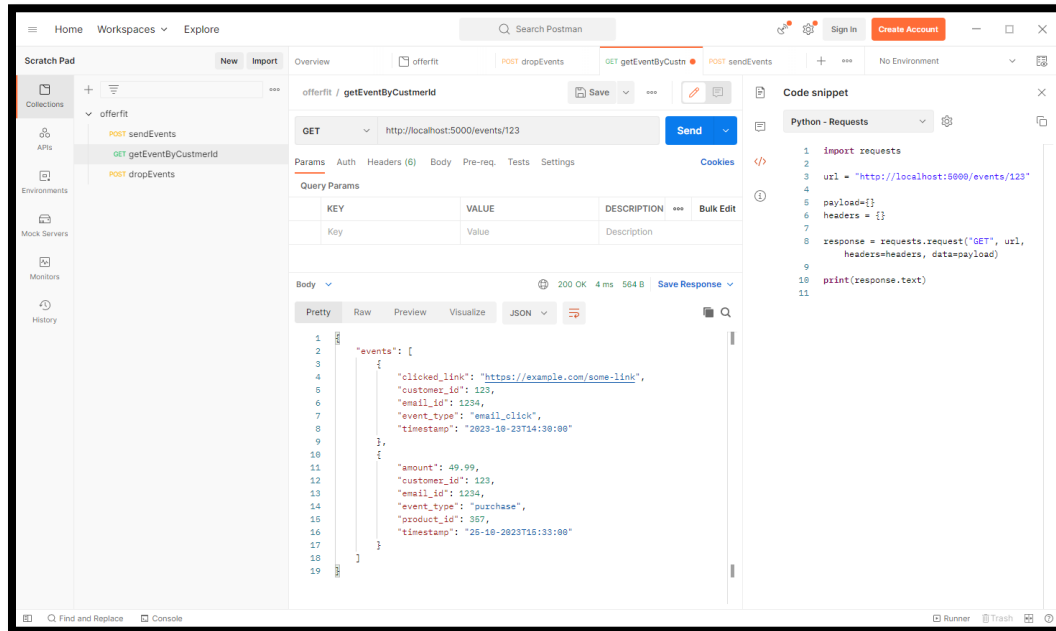
SendEvent (Postman)



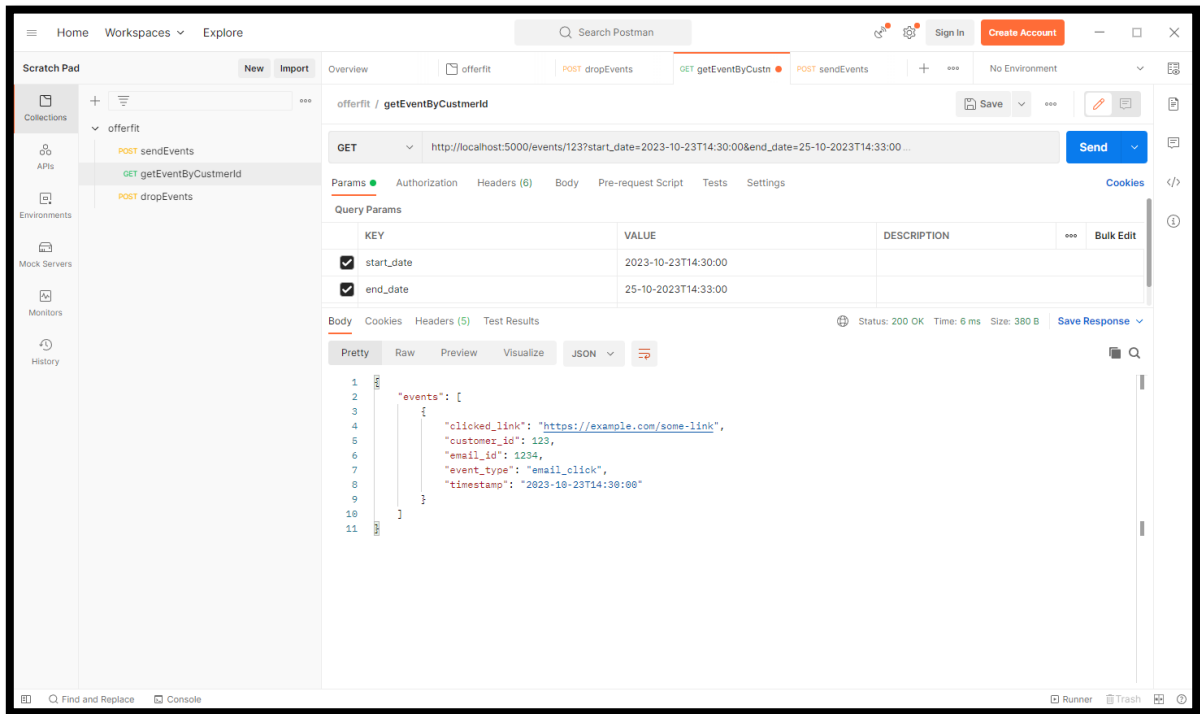
PostgreSQL (table Events)



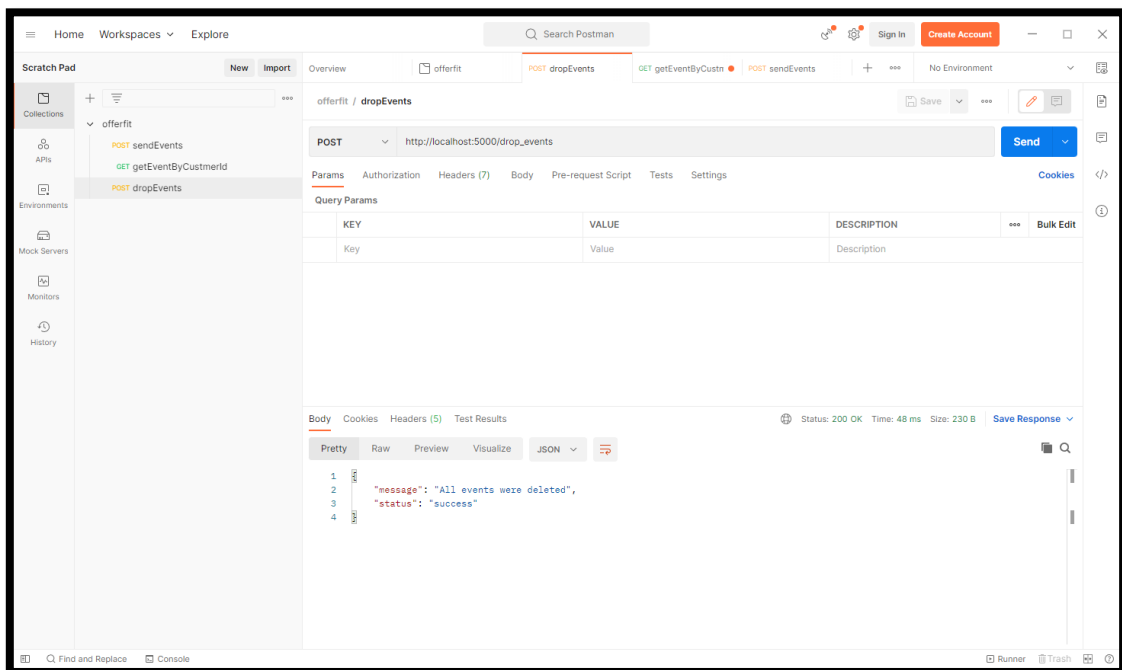
GetEventsByCustomerId (Postman)



GetEventsByCustomerId and filtering by time (Postman):



DropEvents (postman):



PostgreSQL:

