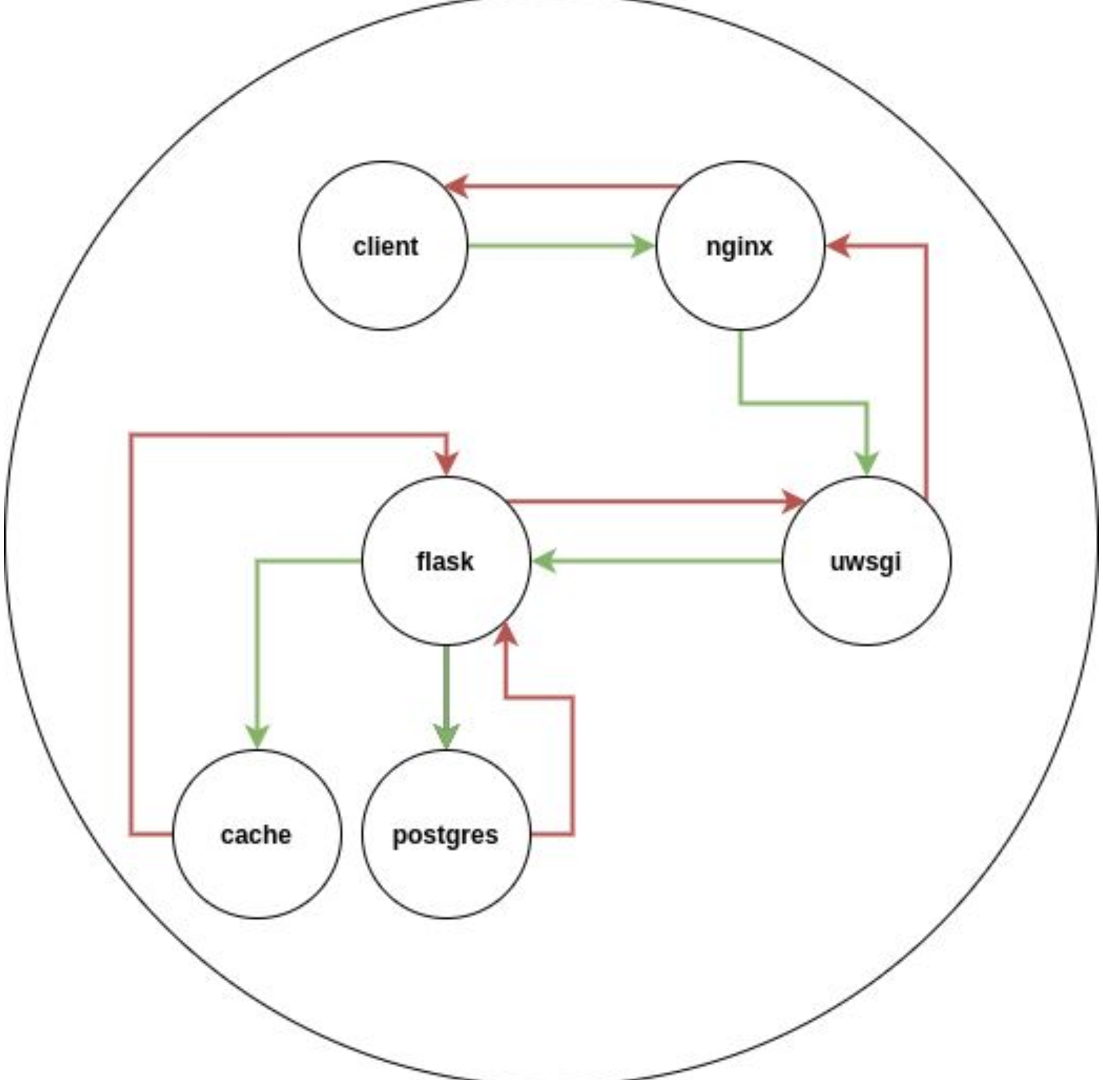
DB Mocking

Blueshell #9

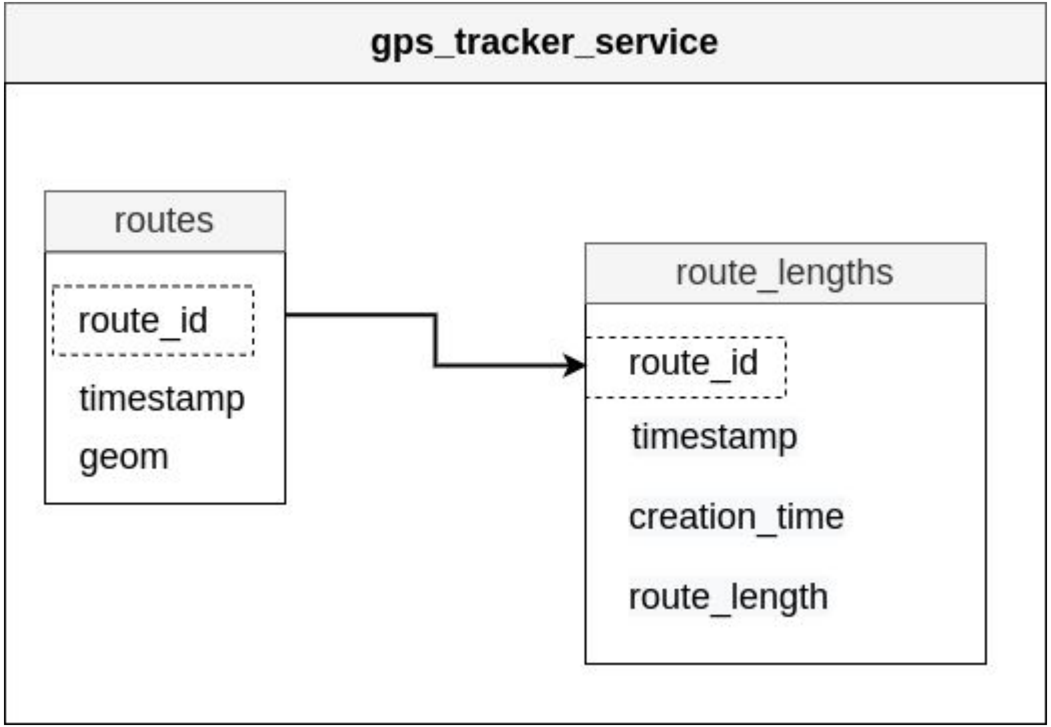
Organization

- views.py - flask endpoints
- controller.py - db logic
- models.py - db access methods
- querys.py - templated sql queries
- cache.py - key/value storage
- test.py - integration tests

Boxology



Application DB: PostGres with PostGis extension



Application Key/Value Cache

```
controller.LONGEST_ROUTE_IN_DAY_CACHE = \

    {YEAR-MONTH-DAY: [long, lat]}
```

Tests

Create a db

- create two tables with schemas

- install postgis extension

Create a new route

- access db for route_id + 1

- if no route_ids, assign route_id 0

Update a route_id with points

- If route_id doesn't exist, create it

Request the length of a route_id

- can include current day

- can include route_ids in progress

Request the longest route for a given date in the past

- only available for previous dates

How to adapt to our needs

- Strip out application logic - Claus/Victor?

- Change table schemas - Claus/Victor?

- wrap the methods to be timed with `timed(lambda: func))` in views.py -

  Dan/David?

- log results to file - Dan/David?

- analyze, discuss - All

- change out db, wash, rinse and repeat