WEATHER DATABASE

WEATHER DATABASE

- Information about weather is very important specially if we want to anticipate weather changes that can affect businesses such as street hawkers, farming, aviation, entertainment etc. Having information about weather beforehand can help people prepare for it to avoid losses.
- For this purpose, it is important to have an application that can let us know about weather alerts and possibly store the weather data in a database for analytics. We present our work on creating a weather forecast app with Python programming language using OpenWeatherMap API and MongoDB.
- It stores the forecast data in MongoDB and displays weather alerts such as snow, rain or freezing temperatures if present in forecast for those cities. It also creates a weather map for each of the forecast and shows weather forecast on a map.

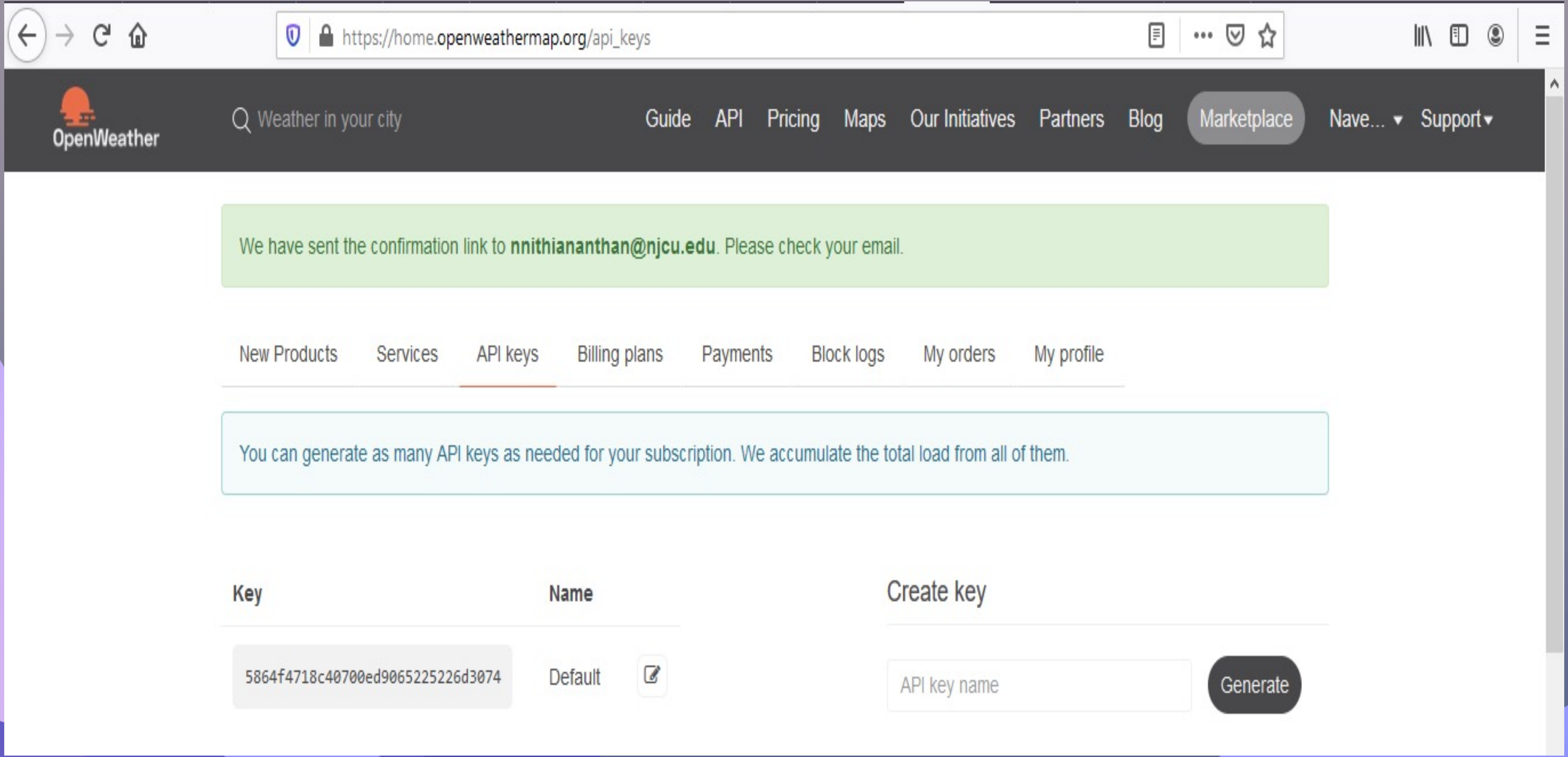
DATASET

OpenWeatherMap is an online service that provides current, historical and weather forecast data for analytics. To communicate with the weather data, user must subscribe to the Openweathermap website at https://openweathermap.org/ and then a user can get API access key. The weather data can be downloaded simply by requesting data from server API endpoint. The data comes in JSON format.

DATA COLLECTION

1. Created an account in https://openweathermap.org/api
2. Got API access key by subscribing to OpenWeatherMap website.
3. Made a multi-threaded program to connect to API. Locations to be monitored would be placed in configuration file.
4. One thread to download 5 days/3-hour forecast.
5. One thread to download weather maps.
6. All data would be stored in database (Mongodb) as separate collections/table.
7. One thread to open the latest weather map and display the map in window (should show last image as per last time stamp).
8. Forecasted threads would print out alerts if there is rain/snow or freezing temperatures (<2-degree Fahrenheit) in any of forecast period.
9. Displayed forecast/previous data from database as a graph.

OpenWeatherMap



https://home.openweathermap.org/api_keys

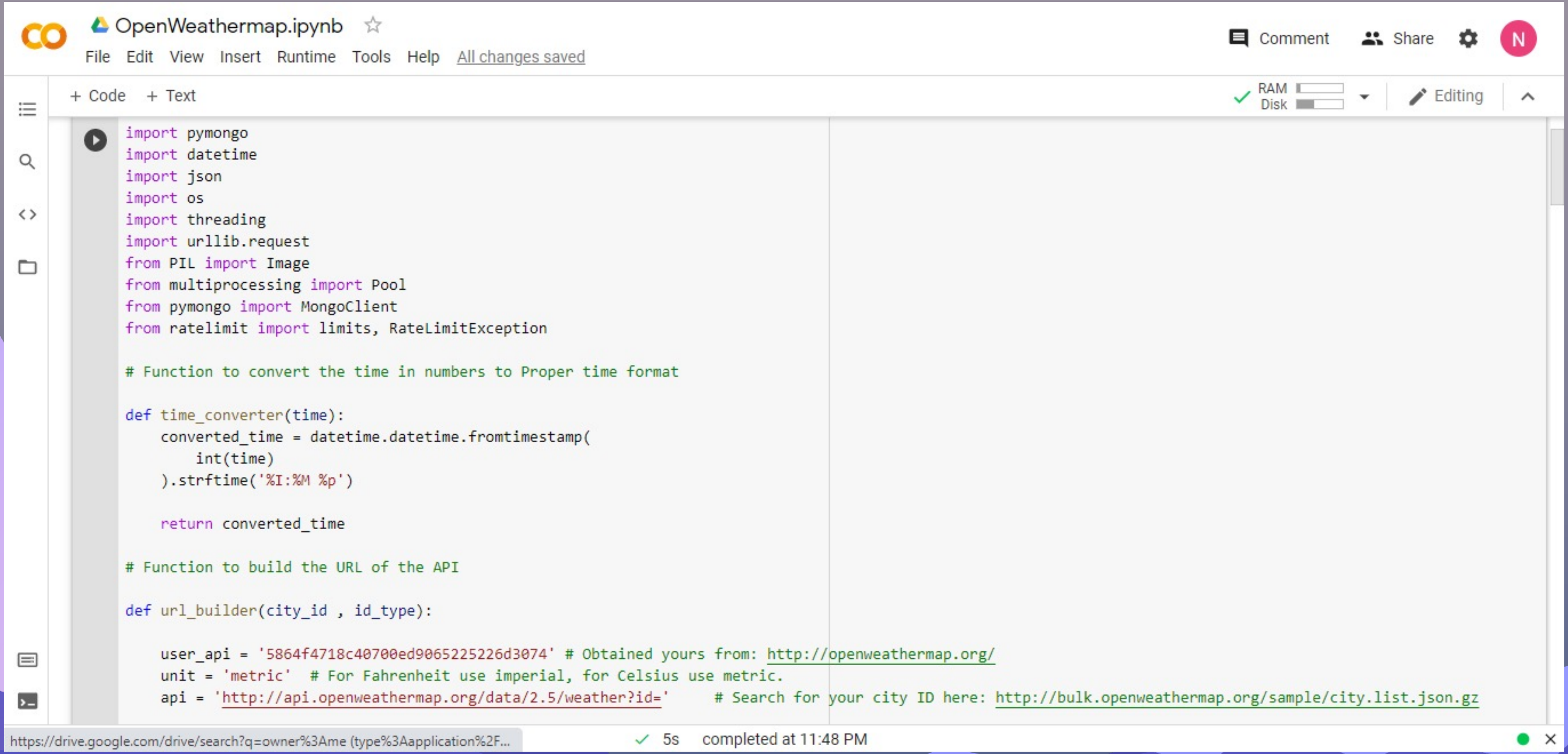
We have sent the confirmation link to **nnithiananthan@njcu.edu**. Please check your email.

You can generate as many API keys as needed for your subscription. We accumulate the total load from all of them.

Key	Name
5864f4718c40700ed9065225226d3074	Default

Create key

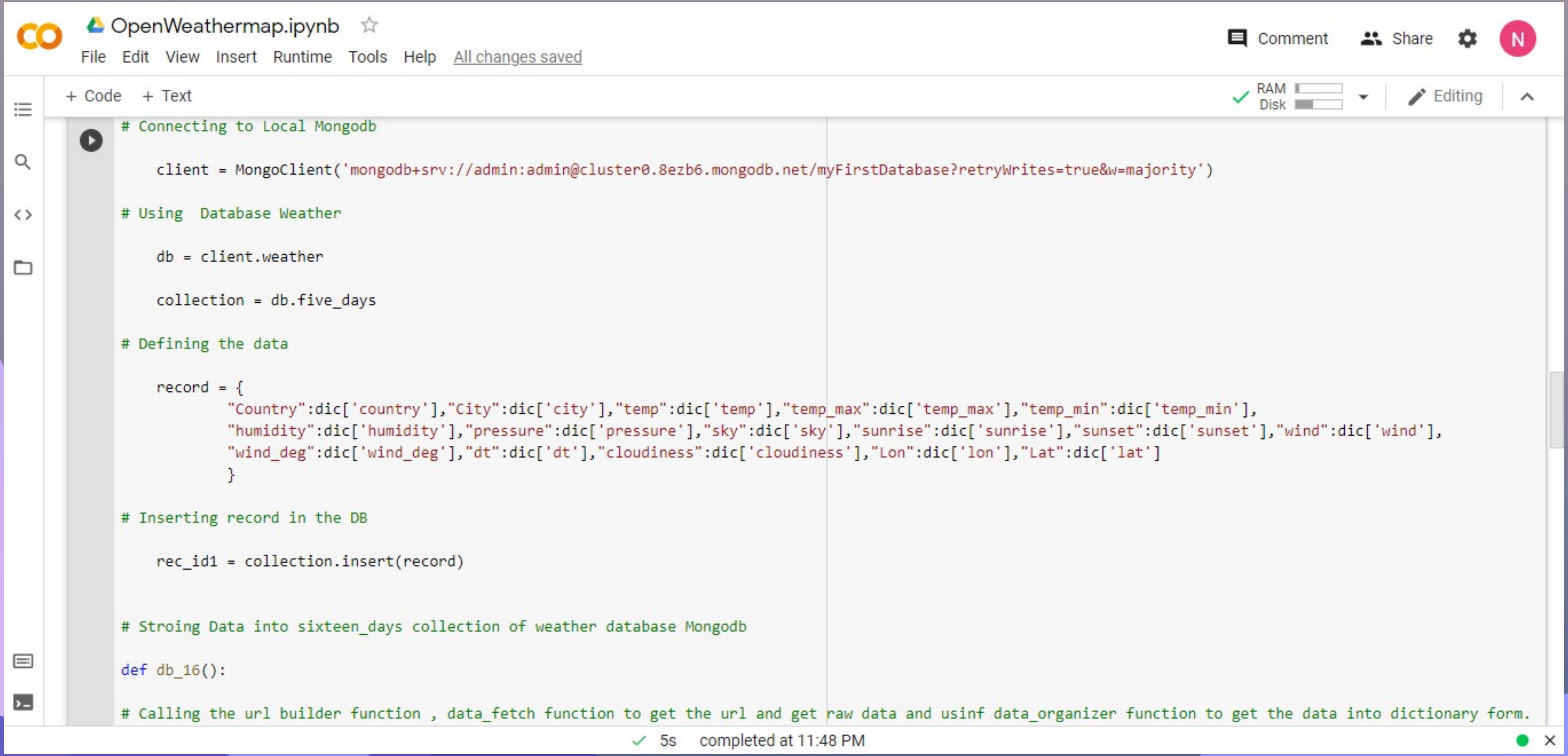
PYTHON –OpenWeatherMap API



```python
import pymongo
import datetime
import json
import os
import threading
import urllib.request
from PIL import Image
from multiprocessing import Pool
from pymongo import MongoClient
from ratelimit import limits, RateLimitException

# Function to convert the time in numbers to Proper time format

def time_converter(time):
    converted_time = datetime.datetime.fromtimestamp(
        int(time)
    ).strftime('%I:%M %p')

    return converted_time

# Function to build the URL of the API

def url_builder(city_id , id_type):

    user_api = '5864f4718c40700ed9065225226d3074' # Obtained yours from: http://openweathermap.org/
    unit = 'metric'  # For Fahrenheit use imperial, for Celsius use metric.
    api = 'http://api.openweathermap.org/data/2.5/weather?id='     # Search for your city ID here: http://bulk.openweathermap.org/sample/city.list.json.gz
```

https://drive.google.com/drive/search?q=owner%3Ame (type%3Aapplication%2F...   ✓ 5s   completed at 11:48 PM

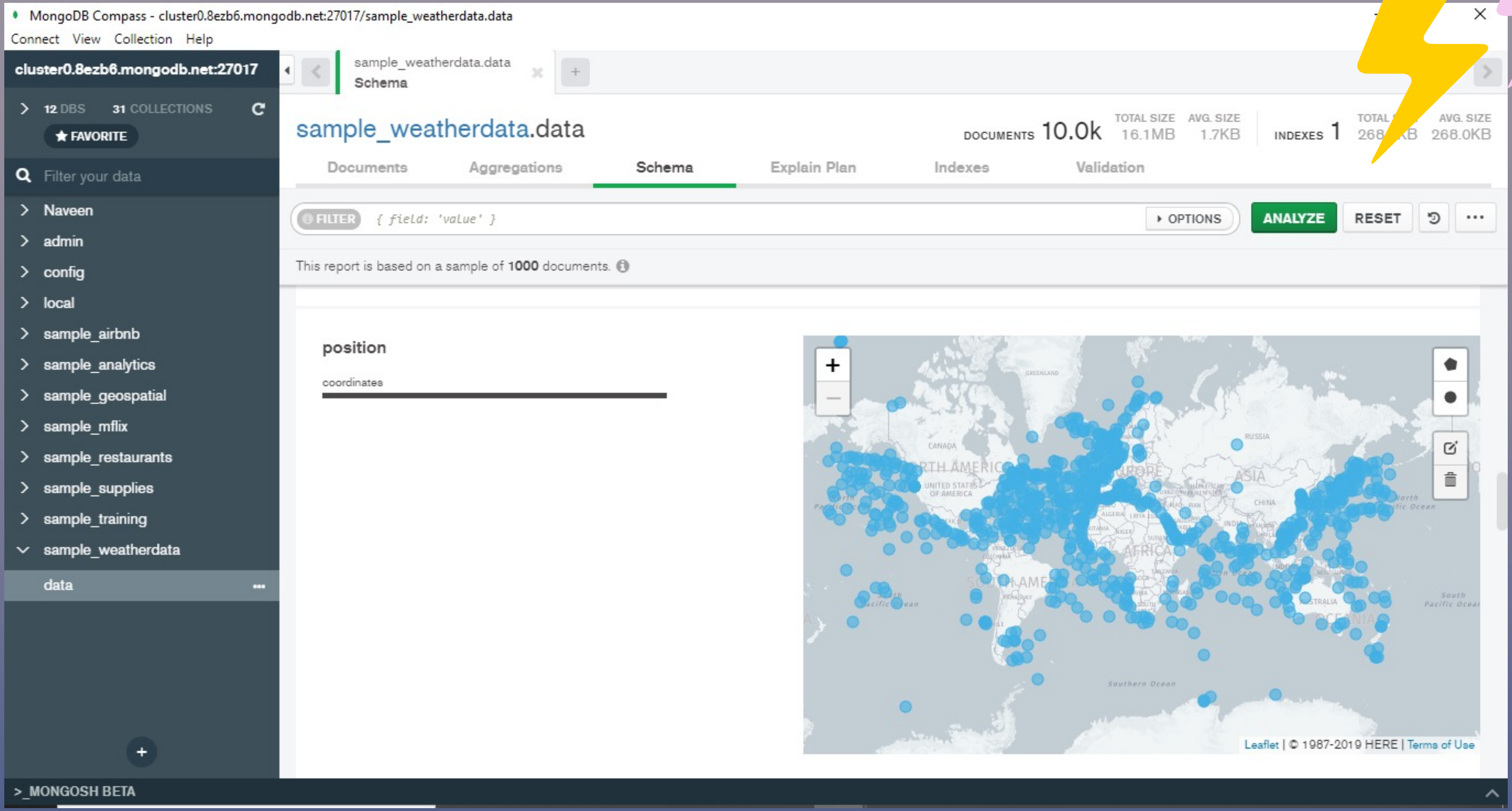
PYTHON –MONGODB API



```
# Connecting to Local Mongodb

    client = MongoClient('mongodb+srv://admin:admin@cluster0.8ezb6.mongodb.net/myFirstDatabase?retryWrites=true&w=majority')

# Using  Database Weather

    db = client.weather

    collection = db.five_days

# Defining the data

    record = {
            "Country":dic['country'],"City":dic['city'],"temp":dic['temp'],"temp_max":dic['temp_max'],"temp_min":dic['temp_min'],
            "humidity":dic['humidity'],"pressure":dic['pressure'],"sky":dic['sky'],"sunrise":dic['sunrise'],"sunset":dic['sunset'],"wind":dic['wind'],
            "wind_deg":dic['wind_deg'],"dt":dic['dt'],"cloudiness":dic['cloudiness'],"Lon":dic['lon'],"Lat":dic['lat']
            }

# Inserting record in the DB

    rec_id1 = collection.insert(record)


# Stroing Data into sixteen_days collection of weather database Mongodb

def db_16():

# Calling the url builder function , data_fetch function to get the url and get raw data and usinf data_organizer function to get the data into dictionary form.
```

```
✓ 5s    completed at 11:48 PM
```

MONGODB COMPASS



MongoDB Compass - cluster0.8ezb6.mongodb.net:27017/sample_weatherdata.data

Connect View Collection Help

cluster0.8ezb6.mongodb.net:27017

12 DBS 31 COLLECTIONS

FAVORITE

Filter your data

- Naveen
- admin
- config
- local
- sample_airbnb
- sample_analytics
- sample_geospatial
- sample_mflix
- sample_restaurants
- sample_supplies
- sample_training
- sample_weatherdata
  - data

sample_weatherdata.data
Schema

sample_weatherdata.data

DOCUMENTS 10.0k TOTAL SIZE 16.1MB AVG. SIZE 1.7KB INDEXES 1 TOTAL SIZE 268.0KB AVG. SIZE 268.0KB

Documents Aggregations Schema Explain Plan Indexes Validation

FILTER { field: 'value' } OPTIONS ANALYZE RESET

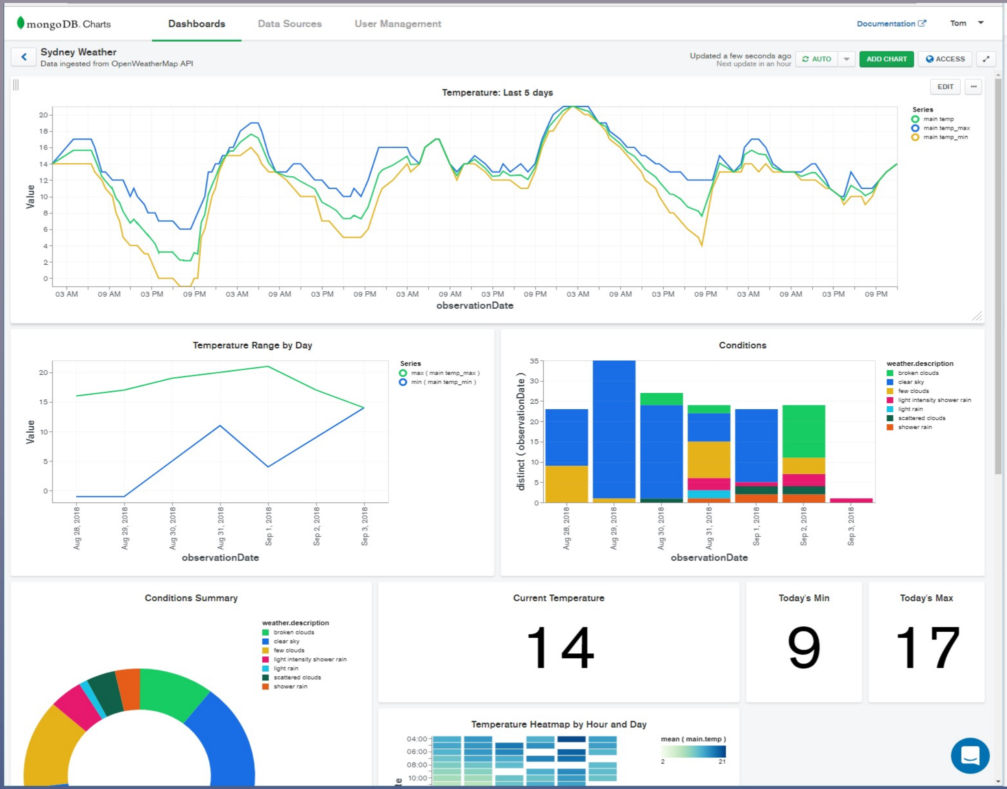
This report is based on a sample of 1000 documents.

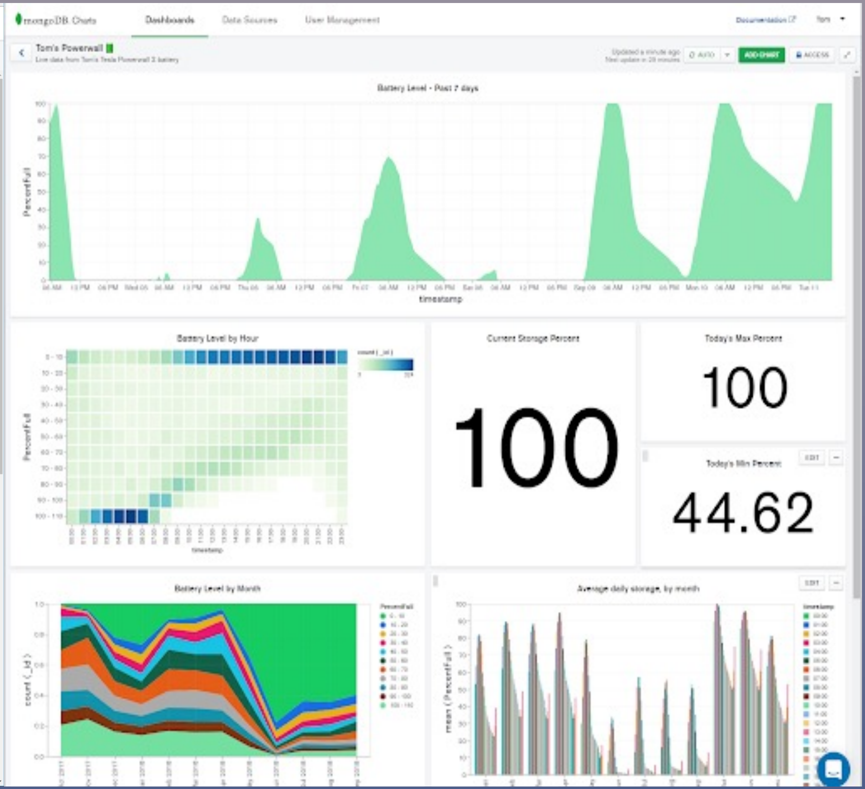
position

coordinates

>_MONGOSH BETA

MONGODB CHARTS



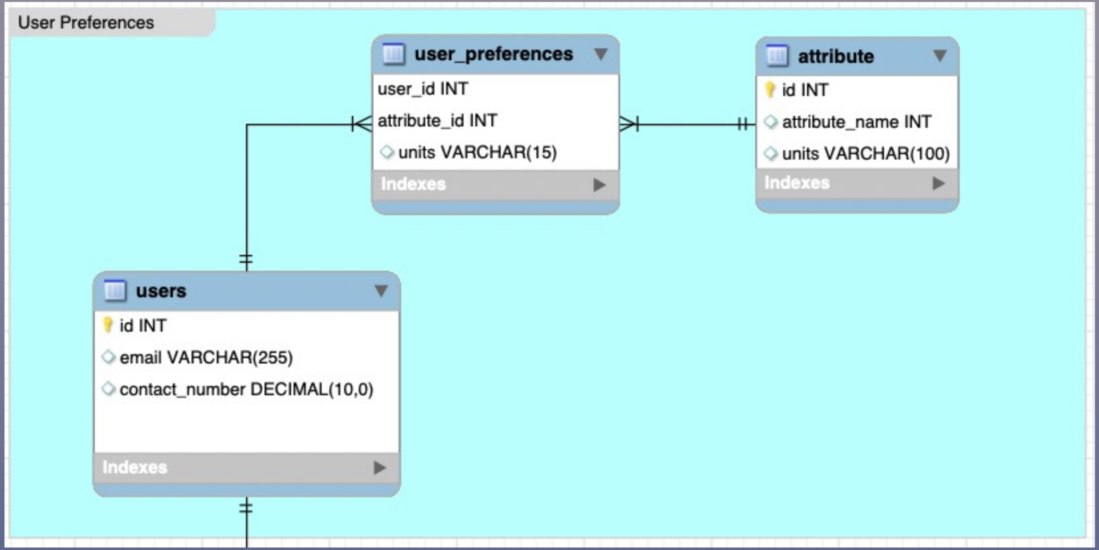


I have split database design into 3 parts:

- User Preference
- User Profile
- Weather Log

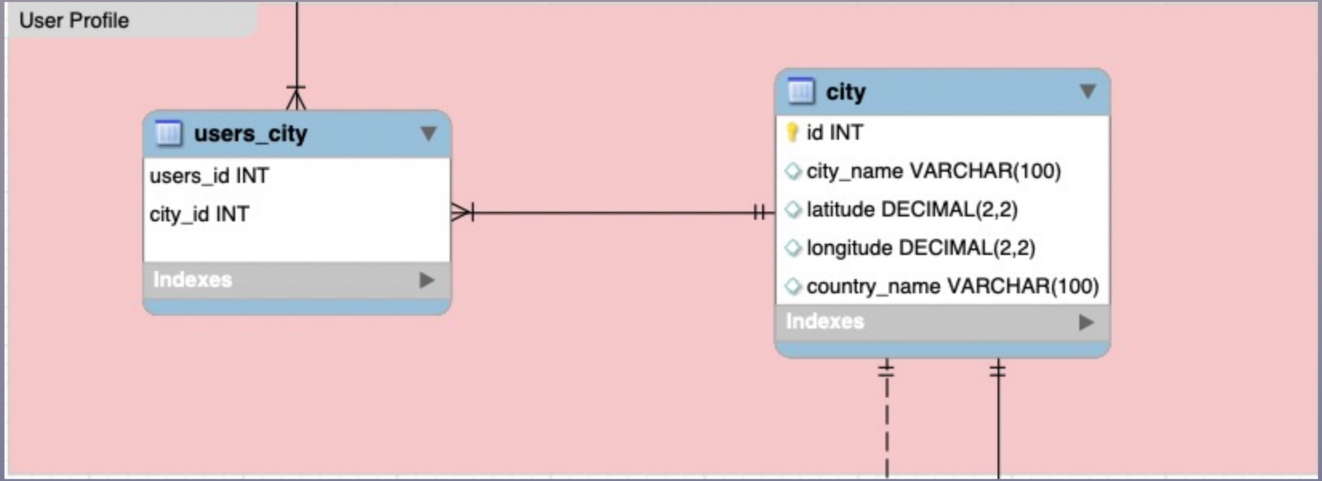
DATABASE DESIGN DIAGRAM

USER PREFERENCE



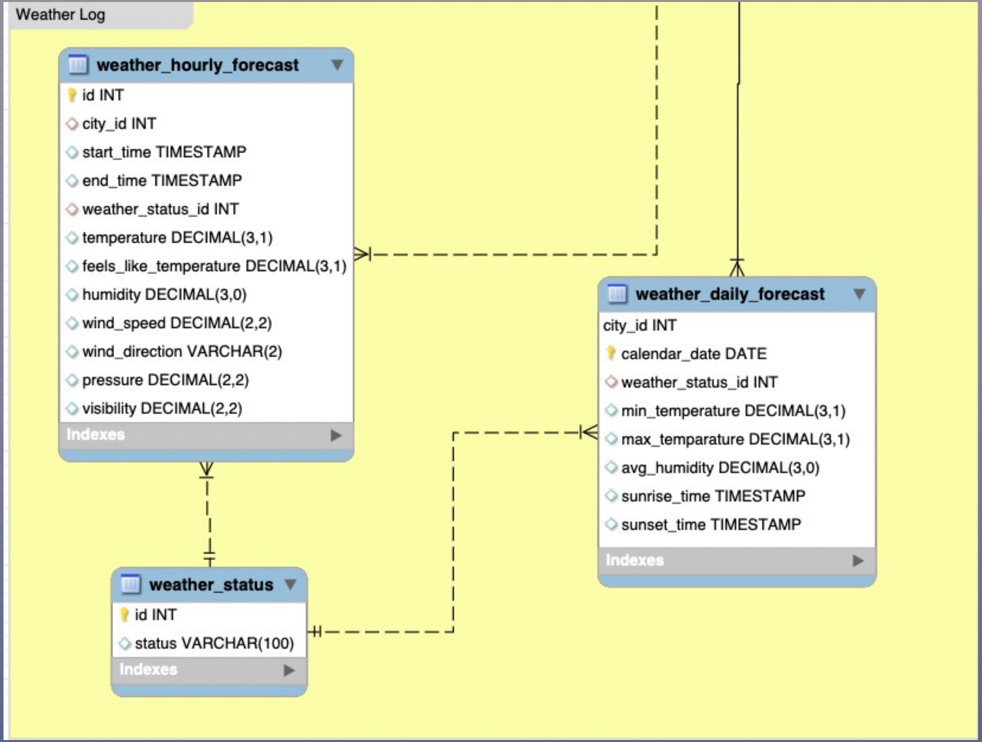
- We will be storing all the users' accounts and user's preference about the unit of measurement.
- For instance, a user may like the temperature to be in Celsius whereas another user may prefer it in Fahrenheit.

USER PROFILE



- we will be storing the city's reference data and user's profile data like what are the cities that a user is interested to get weather alert for.

Weather LOG



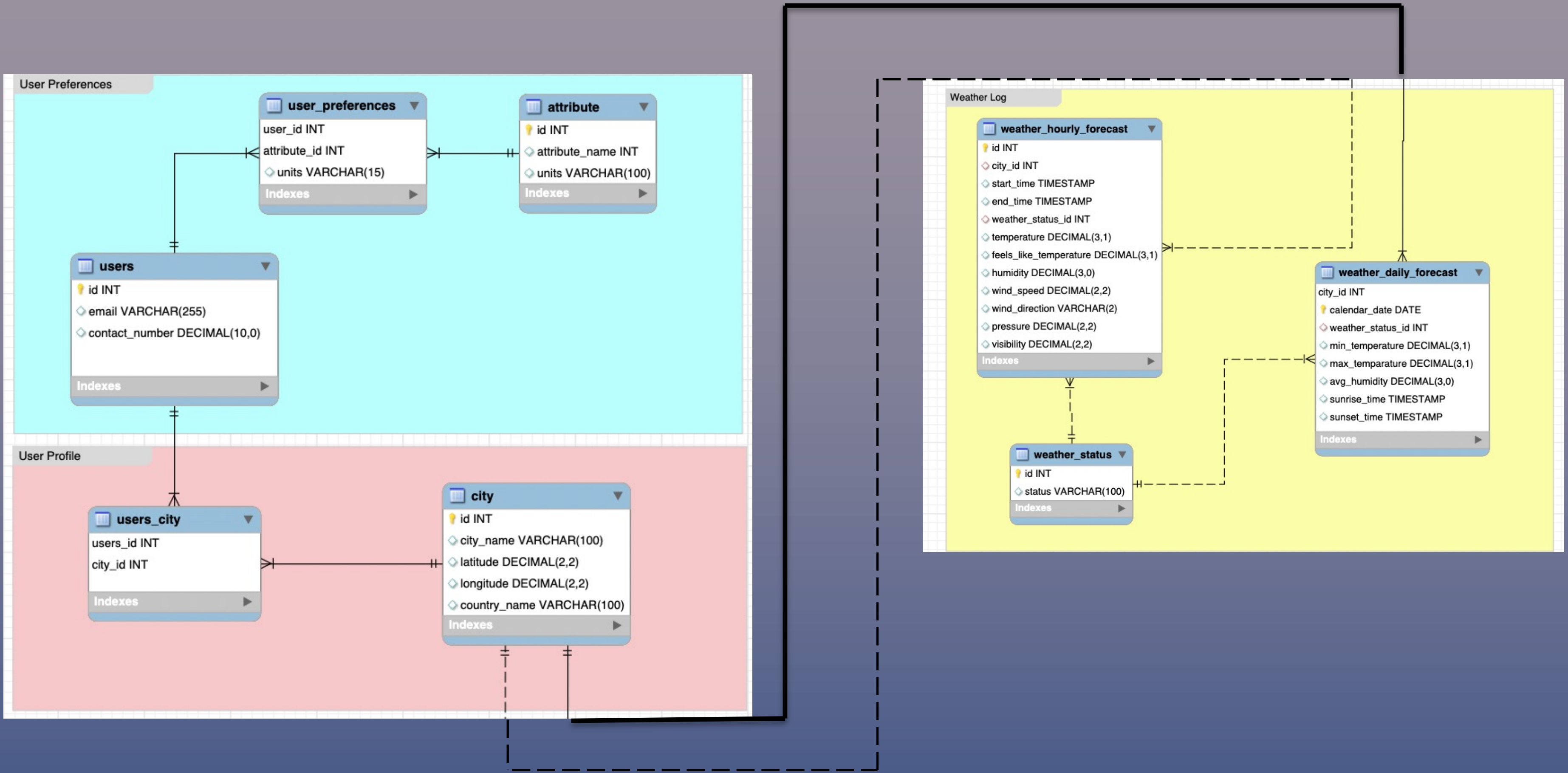
weather_hourly_forecast
- id INT
- city_id INT
- start_time TIMESTAMP
- end_time TIMESTAMP
- weather_status_id INT
- temperature DECIMAL(3,1)
- feels_like_temperature DECIMAL(3,1)
- humidity DECIMAL(3,0)
- wind_speed DECIMAL(2,2)
- wind_direction VARCHAR(2)
- pressure DECIMAL(2,2)
- visibility DECIMAL(2,2)

weather_daily_forecast
- city_id INT
- calendar_date DATE
- weather_status_id INT
- min_temperature DECIMAL(3,1)
- max_temparature DECIMAL(3,1)
- avg_humidity DECIMAL(3,0)
- sunrise_time TIMESTAMP
- sunset_time TIMESTAMP

weather_status
- id INT
- status VARCHAR(100)

we will be storing the weather data in the different tables. We will be storing hourly weather data for every city in weather_hourly_forecast table and daily weather data in weather_daily_forecast table.

DATABASE DESIGN DIAGRAM



PROS AND CONS OF USING RDBMS & NOSQL

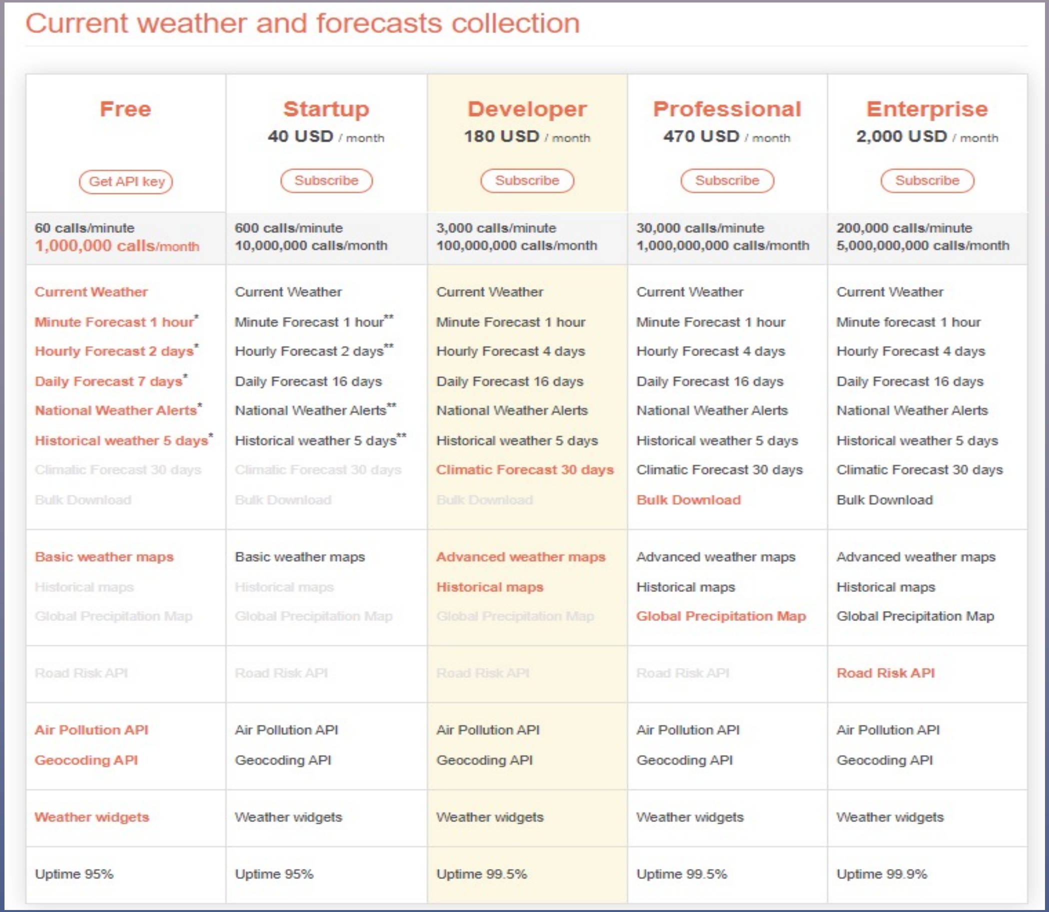
Weather application is read-heavy than write heavy

Volume will be huge considering the weather data for last 10-20 years at hourly level

Mongo DB is better suited for such use case

Will allow us to scale out horizontally with sharding to support huge concurrent number of users

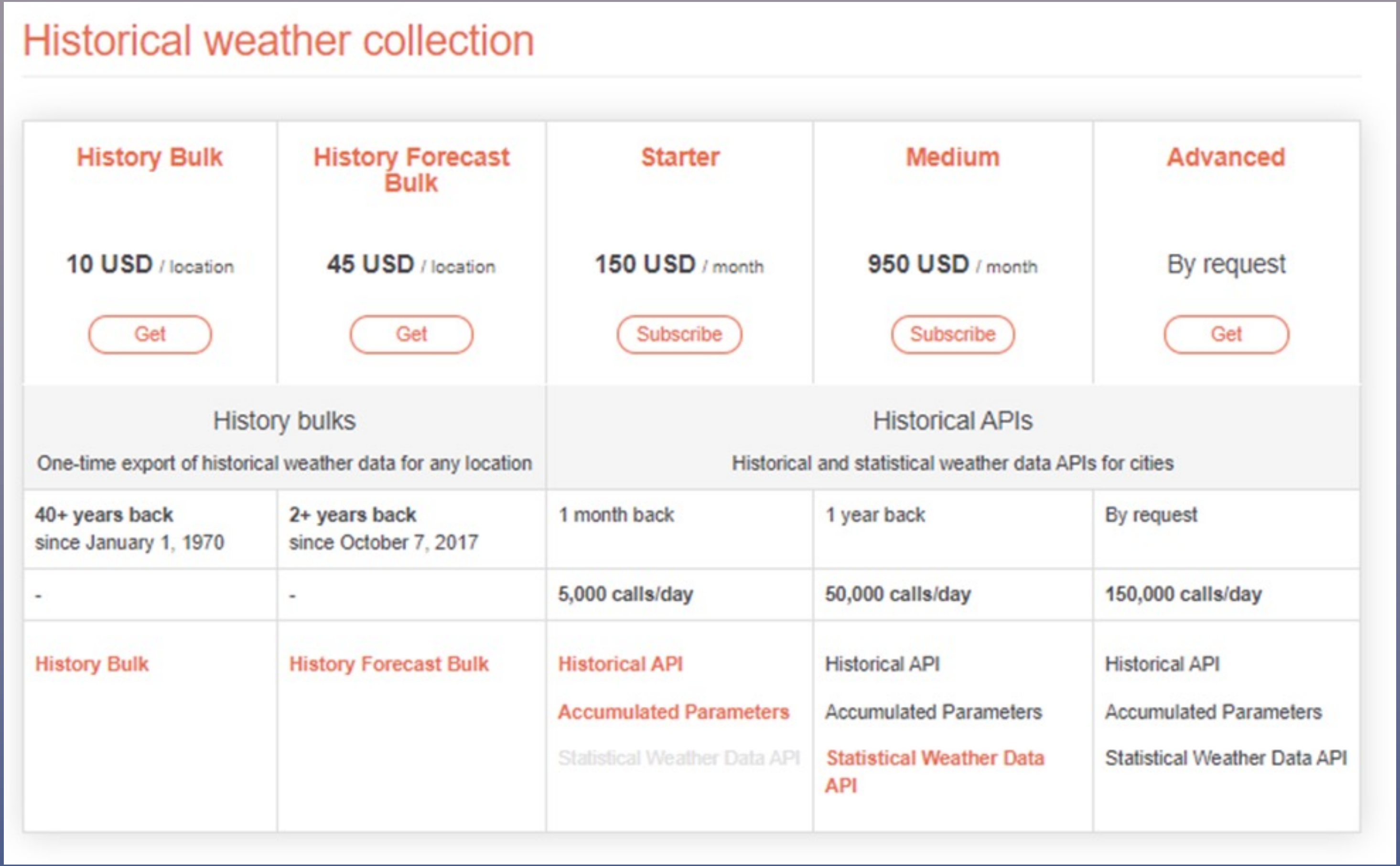
COST OF DATA COLLECTION



Current weather and forecasts collection

Free	Startup	Developer	Professional	Enterprise
	40 USD / month	180 USD / month	470 USD / month	2,000 USD / month
Get API key	Subscribe	Subscribe	Subscribe	Subscribe
60 calls/minute	600 calls/minute	3,000 calls/minute	30,000 calls/minute	200,000 calls/minute
1,000,000 calls/month	10,000,000 calls/month	100,000,000 calls/month	1,000,000,000 calls/month	5,000,000,000 calls/month
Current Weather	Current Weather	Current Weather	Current Weather	Current Weather
Minute Forecast 1 hour*	Minute Forecast 1 hour**	Minute Forecast 1 hour	Minute Forecast 1 hour	Minute forecast 1 hour
Hourly Forecast 2 days*	Hourly Forecast 2 days**	Hourly Forecast 4 days	Hourly Forecast 4 days	Hourly Forecast 4 days
Daily Forecast 7 days*	Daily Forecast 16 days	Daily Forecast 16 days	Daily Forecast 16 days	Daily Forecast 16 days
National Weather Alerts*	National Weather Alerts**	National Weather Alerts	National Weather Alerts	National Weather Alerts
Historical weather 5 days*	Historical weather 5 days**	Historical weather 5 days	Historical weather 5 days	Historical weather 5 days
Climatic Forecast 30 days	Climatic Forecast 30 days	Climatic Forecast 30 days	Climatic Forecast 30 days	Climatic Forecast 30 days
Bulk Download	Bulk Download	Bulk Download	Bulk Download	Bulk Download
Basic weather maps	Basic weather maps	Advanced weather maps	Advanced weather maps	Advanced weather maps
Historical maps	Historical maps	Historical maps	Historical maps	Historical maps
Global Precipitation Map	Global Precipitation Map	Global Precipitation Map	Global Precipitation Map	Global Precipitation Map
Road Risk API	Road Risk API	Road Risk API	Road Risk API	Road Risk API
Air Pollution API	Air Pollution API	Air Pollution API	Air Pollution API	Air Pollution API
Geocoding API	Geocoding API	Geocoding API	Geocoding API	Geocoding API
Weather widgets	Weather widgets	Weather widgets	Weather widgets	Weather widgets
Uptime 95%	Uptime 95%	Uptime 99.5%	Uptime 99.5%	Uptime 99.9%

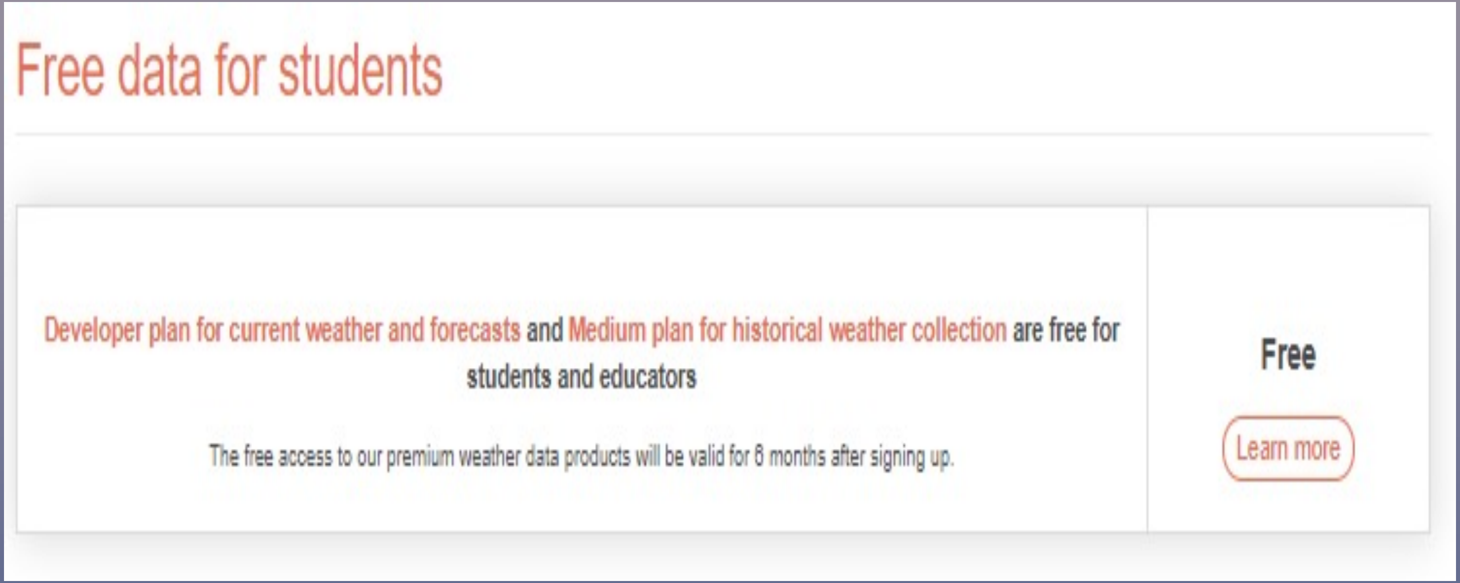
Contd..



Historical weather collection

History Bulk	History Forecast Bulk	Starter	Medium	Advanced
10 USD / location	45 USD / location	150 USD / month	950 USD / month	By request
Get	Get	Subscribe	Subscribe	Get
History bulks One-time export of historical weather data for any location		Historical APIs Historical and statistical weather data APIs for cities		
40+ years back since January 1, 1970	2+ years back since October 7, 2017	1 month back	1 year back	By request
-	-	5,000 calls/day	50,000 calls/day	150,000 calls/day
History Bulk	History Forecast Bulk	Historical API Accumulated Parameters Statistical Weather Data API	Historical API Accumulated Parameters Statistical Weather Data API	Historical API Accumulated Parameters Statistical Weather Data API

Contd..



Free data for students

Developer plan for current weather and forecasts and Medium plan for historical weather collection are free for students and educators

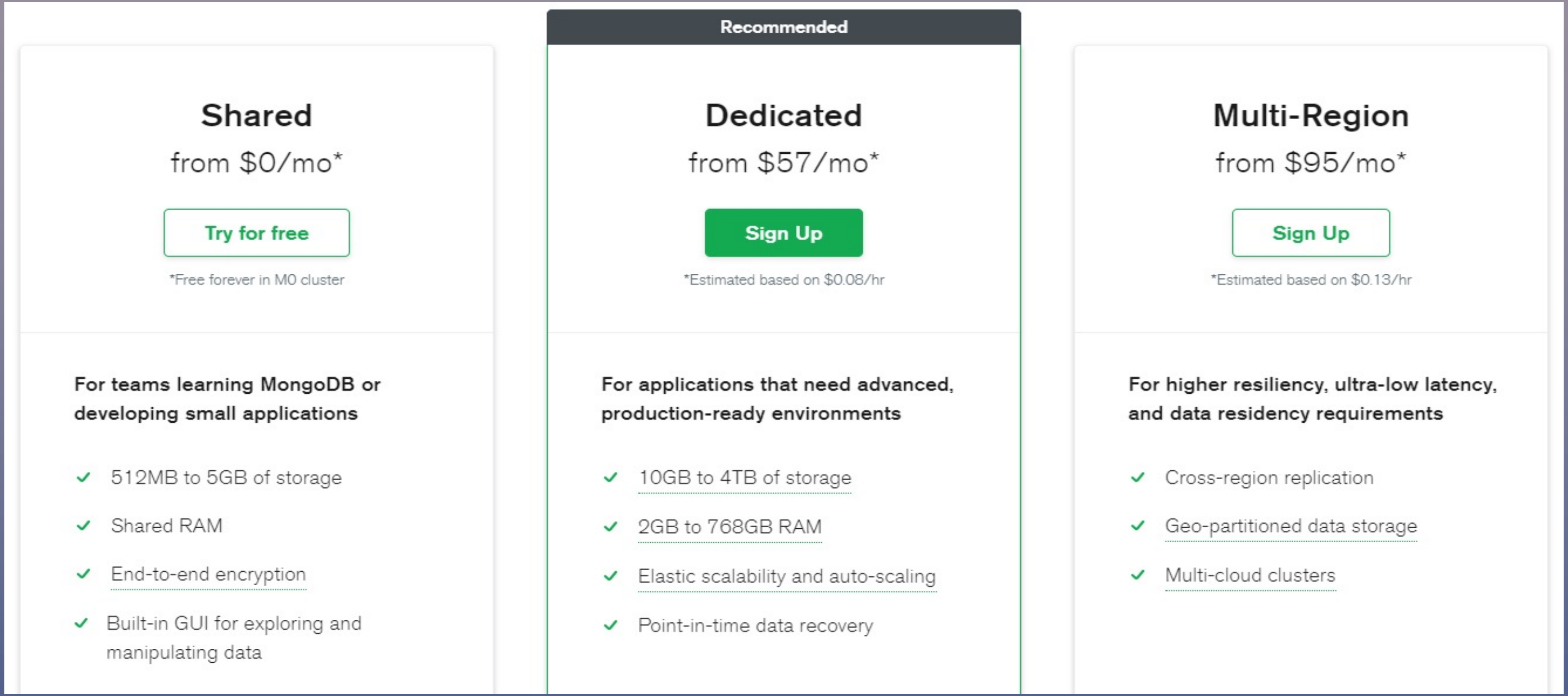
The free access to our premium weather data products will be valid for 6 months after signing up.

Free

Learn more

CLOUD STORAGE

- Price Details for MongoDB Subscription



Recommended

Shared

from $0/mo*

*Free forever in M0 cluster

For teams learning MongoDB or developing small applications

- ✓ 512MB to 5GB of storage
- ✓ Shared RAM
- ✓ End-to-end encryption
- ✓ Built-in GUI for exploring and manipulating data

Dedicated

from $57/mo*

*Estimated based on $0.08/hr

For applications that need advanced, production-ready environments

- ✓ 10GB to 4TB of storage
- ✓ 2GB to 768GB RAM
- ✓ Elastic scalability and auto-scaling
- ✓ Point-in-time data recovery

Multi-Region

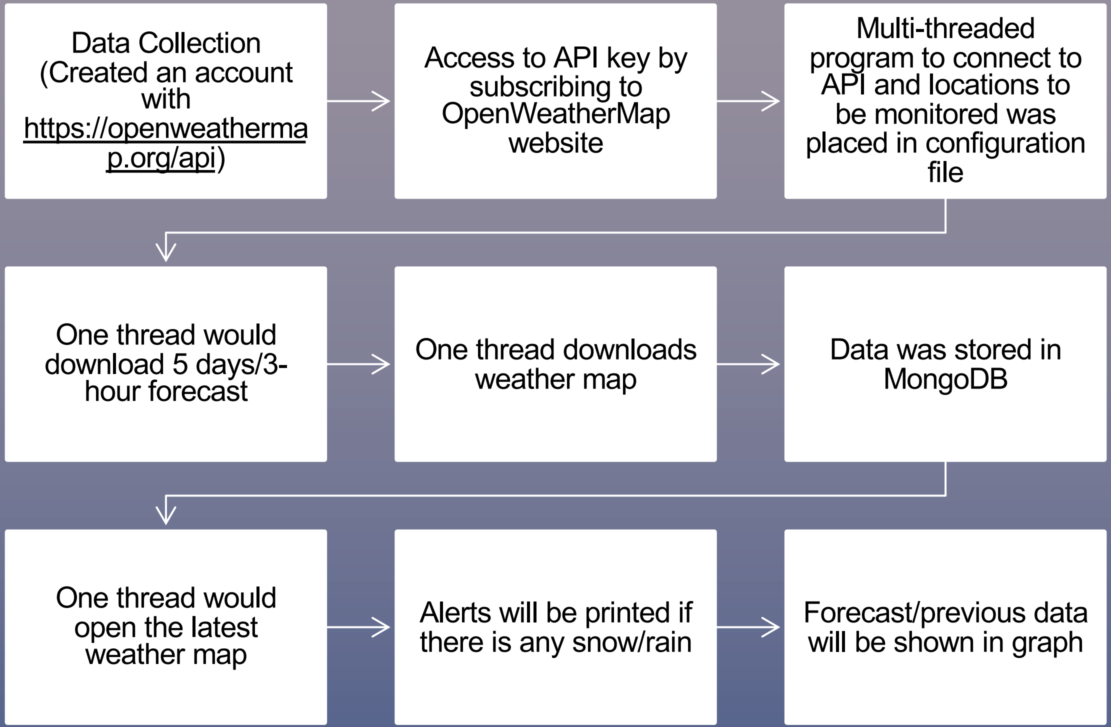
from $95/mo*

*Estimated based on $0.13/hr

For higher resiliency, ultra-low latency, and data residency requirements

- ✓ Cross-region replication
- ✓ Geo-partitioned data storage
- ✓ Multi-cloud clusters

TIMELINE OF PROJECT



Data Collection (Created an account with https://openweathermap.org/api)

Access to API key by subscribing to OpenWeatherMap website

Multi-threaded program to connect to API and locations to be monitored was placed in configuration file

One thread would download 5 days/3-hour forecast

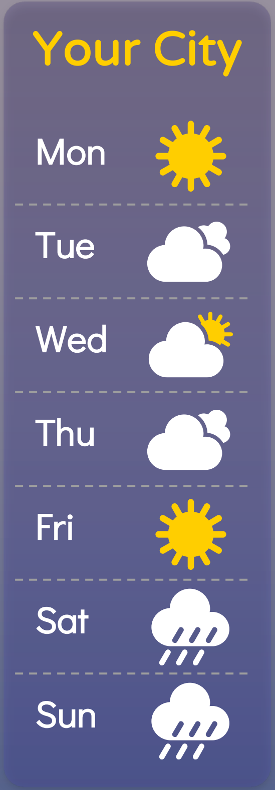
One thread downloads weather map

Data was stored in MongoDB

One thread would open the latest weather map

Alerts will be printed if there is any snow/rain

Forecast/previous data will be shown in graph



THANKS!