# Basic Rate Limiter

## What is Rate Limit?

Rate limiting is a strategy for limiting network traffic. It puts a cap on how often someone can repeat an action within a certain timeframe. Rate limiting can help stop certain kinds of malicious bot activity. It can also reduce strain on web servers.

Uses of Rate Limiter:

- Rate limiting is often employed to stop bad bots from negatively impacting a website or application. Bot attacks that rate limiting can help mitigate include:
- Brute force attacks
- DoS and DDoS attacks
- Web scraping
- Rate limiting also protects against API overuse, which is not necessarily malicious or due to bot activity, but is important to prevent nonetheless.

## Approach

I have designed the app in Django and I have also created the rate limiter default as 100 and rate-limit for each user with each endpoint

I have used Database cache features in Django.
It stores the data as

```
{
username: {
          ‘journey’: 10,
          ‘boundary: 10
                    }
          }
```

The Data Structure stores the rate-limit data per-user, per service-type (journey, boundary)

There are following endpoints that have specific work.

| ENDPOINT | DESCRIPTION |
|---|---|
| Journey/ | Its shows the rate-limit for logged in user, in journey type service |
| boundary/ | Its shows the rate-limit per user, in boundary type service |
| journey/reset | It's reset the rate-limit user logged in, in journey type service |
| boundary/reset | It's reset the rate-limit user logged in, in boundary type service |
| clear | It's reset the rate-limit of all user present in cache |
| users | It's showing the list of users' objects |
| register | It helps in creating new user with rate-limit |

Limitation: We need to create rate-limit on admin dashboard

# Improvements

We can use **time stamps** for later improvement. We can use various other method to figure out the time-based rate-limit per user or collective in total

We can also use **session based** or **IP based** user rate-limit.
We can keep the time frame smaller and for further implementation of rate limit we can **choose hash map** or **Sliding window technique**

For the further enhancement is should Redis as cache memory instead of inbuilt Database Cache.

We could have also chosen Flask instead of Django and Django Rest Framework

We can also use token based authentication